

4|86 DAS MAGAZIN FÜR COMPUTER-FANS

C 64: Steuermann und Meßknecht

Grundlagen und Beispiele: Messen, Steuern, Regeln

Neue Serie: Der Basic-Baukasten zum Abtippen

Super-Platinen im Selbstbau

Profi-Spiele

Ganz leicht selbst gemacht
Kompletter Test und großer Wettbewerb

Mit vielen ergänzenden Informationen zur Fernsehserie
COMPUTERZEIT
Messen, Steuern und Regeln



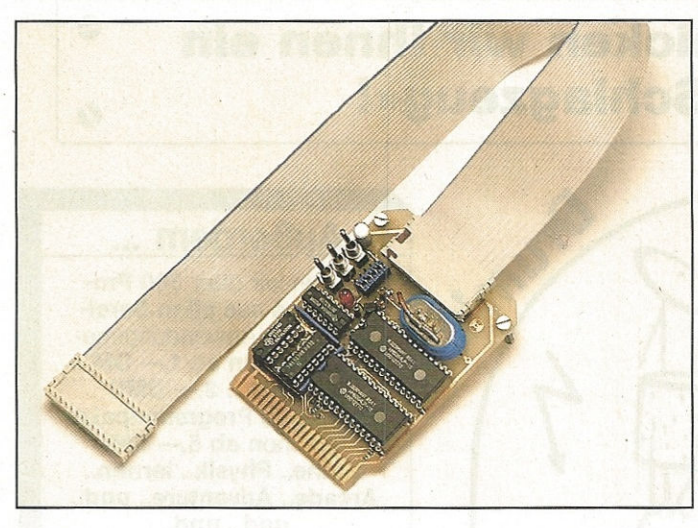


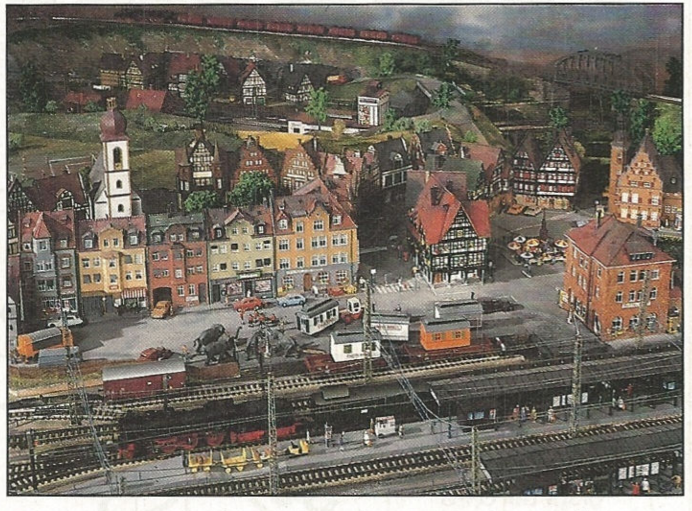


INHALT 

CMOS-RAM-Platine

Jetzt ist es kein Problem mehr, ein nach eigenen Vorstellungen modifiziertes Betriebssystem für den Computer herzustellen oder dem Drucker einen neuen Zeichensatz zu verpassen. Ein Programm bleibt jetzt auch bei ausgeschaltetem Computer erhalten. Änderungen können Sie aber jederzeit vornehmen.



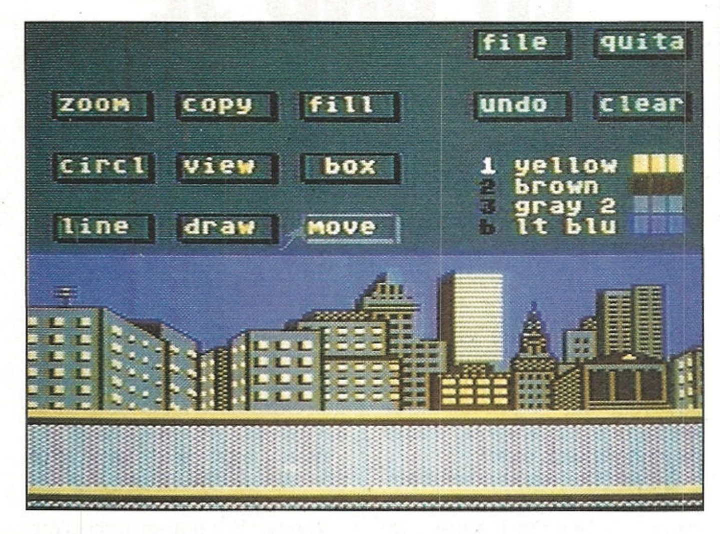


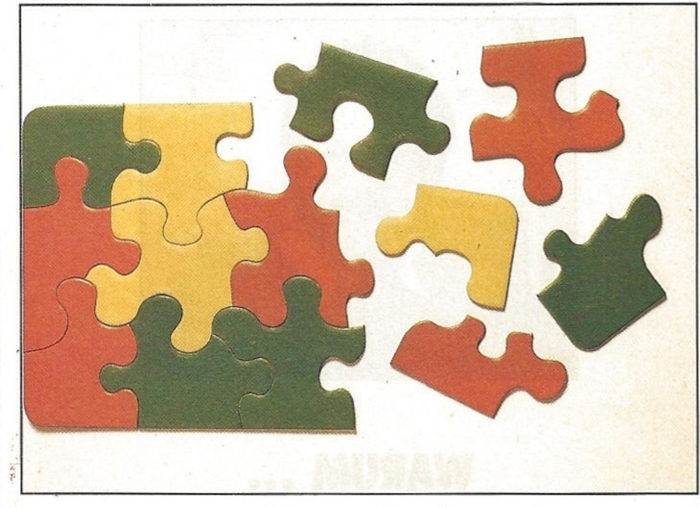
Messen — Steuern — Regeln

Der Computer übernimmt die Kontrolle über seine Umwelt. Auf der Modelleisenbahn fahren die Züge jetzt programmgesteuert. Der Roboter greift gezielt nach einem Werkstück. Der C 64 wird zu einem Oszilloskop. Um das zu verwirklichen, muß Ihr Computer Messen — Steuern — Regeln. Wir zeigen Ihnen, wie es gemacht wird.

GameMaker

Wem das Selberprogrammieren von Spielen bisher zu schwer war, kann auf Hilfe hoffen. Der GameMaker von Activision soll Sie um die gefährlichsten Klippen der Spieleprogrammierung herumschiffen. GameMaker ist ein sehr komfortabler Spielegenerator für Bild und Musik.

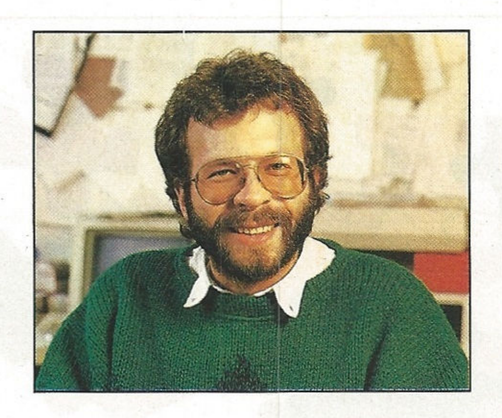




Hypra-Basic

Erstellen Sie sich Ihre eigenen Basic-Erweiterungen mit Hypra-Basic. Durch seinen modularen Aufbau erlaubt es jede beliebige Zusammenstellung von Befehlen. Weiterhin können eigene Routinen eingebunden und später als Basic-Befehl aufgerufen werden. Somit können Sie Hypra-Basic Ihren persönlichen Bedürfnissen anpassen.

▪ Dieses Symbol zeigt an, welche Programme auf Diskette erhältlich sind.



WARUM ...

sieht die 64'er diesmal im Innern ganz anders aus als sonst?

Aus den Umfragen der Vergangenheit wissen wir, was einem Großteil unserer Leser in bezug auf die optische Gestaltung der 64'er gefällt, beziehungsweise zu dem damaligen Zeitpunkt gefallen hat. Aus diesem Wissen heraus haben wir mehrere Entwürfe für das zukünftige Aussehen der 64'er angefertigt und verschiedenen Lesergruppen zur Stellungnahme vorgelegt. Was dabei gut ankam, sehen Sie auf den folgenden Seiten. Wir hoffen, daß es Ihnen allen gefällt und daß Sie uns Ihre Meinung dazu schreiben — egal ob Kritik, Zustimmung, Ablehnung oder weitere Anregungen. Wir freuen uns über jede Zuschrift.

Wir verfolgen mit dieser neuen Gestaltung noch einen weiteren Zweck: Wir sparen Platz. Mit anderen Worten heißt das, daß wir mit der gleichen Seitenzahl mehr Informationen unterbringen. Das heißt für Sie, unsere Leser, daß Sie gleichviel oder sogar mehr wertvolle Informationen bekommen und zwar zum gleichen Preis. Denn auch wir bleiben von Kostensteigerungen nicht verschont.

Mit dieser Maßnahme haben wir in Ihrem Interesse zwei Dinge erreicht: Wir müssen in nächster Zeit entweder keine Preiserhöhung vornehmen oder falls es später wirklich einmal sein muß, wird diese niedriger ausfallen.

Doch nun zu einem anderen Punkt: Überall liest und hört man, daß Commodore Probleme hat. Die wildesten Gerüchte kommen in Umlauf — und das Schlimmste: Es wird nicht nur daran geglaubt, sondern die Gerüchte vermehren sich auch noch. Dies war für uns der Anlaß, mit dem wichtigsten Commodore-Mann in Europa, Harald Speyer, Vice President Europe Commodore International, zu sprechen. Wie er die Lage beurteilt, wie er die Zukunft von Commodore sieht, was an den Gerüchten wahr ist und was nicht, lesen Sie auf den nächsten Seiten. Ich möchte betonen, daß wir nicht das Sprachrohr von Commodore sind, daß wir es aber als unsere Pflicht ansehen, Commodore einmal Gelegenheit zu geben, zu den ganzen Gerüchten der letzten Zeit, zu den sicherlich vorhandenen Problemen, aber auch zur Zukunft von Commodore Stellung zu nehmen.

Michael Scharfenberger, Chefredakteur

COMMODORE — STARK WIE EH UND JE

Der Commodore 64 ist mit 6 Millionen Stück der meistverkaufteste Computer aller Zeiten. Dennoch schien der Hersteller in Schwierigkeiten geraten zu sein. Ein Gespräch mit dem Vice President von Commodore International, Harald Speyer, schafft Klarheit.

64'er: Commodore ist in der letzten Zeit ziemlich häufig in die Schlagzeilen gekommen, unter anderem war sogar die Rede von Zahlungsunfähigkeit. Was ist da jetzt wirklich dran?

Speyer: Das sind zunächst einmal Gerüchte, die ihren Ursprung in New York hatten, wo vielleicht irgendein Finanzgenie von eigenen Gnaden die Termine, die wir mit den Banken hatten und unsere Kreditlinien betrafen, verfolgt hatte, und dann Ende Januar zu dem Schluß kam, wenn Commodore morgen, am 31. Januar, seine Kredite nicht zurückzahlt, geht Commodore unter Chapter 11 (Chapter 11 ist ein Schutzparagraph in den USA, der zahlungsunfähige Unternehmen vor den Gläubigern schützt; Anm. d. Red.). Das wurde dann im Wall Street Journal, um es beim Namen zu nennen, veröffentlicht. Ich weiß nicht, ob diese Wirtschaftsjournalisten, die für Wall Street schreiben, überhaupt etwas von Wirtschaft verstehen. Es scheint mir so, daß sie dort sehr unbedarft sind. Aus einem Grund: Die wenigsten Unternehmen der Welt können ihre Kredite, die sie von Banken haben, von heute auf morgen zurückzahlen. Wenn das gefordert würde, wären 85 Prozent aller Unternehmen von heute auf morgen zahlungsunfähig. Zum anderen wäre eine Branche, die es seit Urzeiten gibt, nicht mehr existent, die Banken. Die leben nämlich vom Geldverleih.

Zu der Frage, was war an den Gerüchten dran: Wir hatten aus der Vergangenheit relativ hohe Kreditlinien. Die Firma ist in den letzten zwei Jahren durch einen Prozeß der Veränderung gegangen. Wir haben unsere Kreditlinien vereinbarungsgemäß reduziert. So wurden im letzten Quartal über 50 Millionen Dollar zurückgezahlt. Der Umsatz konnte leicht gesteigert werden. Wir haben einen besseren Cash Flow (Brutto-Überschuß, Nettoeinnahme; Anm. d. Red.) Ende Dezember 1985 als Ende Dezember 1984. Insgesamt ist an den Gerüchten nicht mehr dran, als daß es Gerüchte waren.

64'er: Sind jetzt tatsächlich Spezialvereinbarungen mit den Banken getroffen worden?

Speyer: Wir haben immer Vereinbarungen mit den Banken gehabt, wie es jedes Unternehmen hat. Wir haben Kreditlinien in allen Ländern wo wir tätig sind. Wir haben allerdings noch nie einen Journalisten vom Wall Street Journal zu unseren Verhandlungen eingeladen, weder in der Vergangenheit, noch in der Gegenwart, noch werden wir das in der Zukunft tun, noch werden wir Details der Vereinbarung mit den Banken mit diesen Leuten absprechen. Diese sogenannten Fachjournalisten der Wirtschaftspresse haben sich schlicht und einfach auf etwas eingeschossen, das jeder vernünftig denkende Mensch ad absurdum führen kann. Und das Schlimme dabei ist, der eine schreibt vom anderen ab, ohne selbst zu recherchieren. Wenn im Wall Street Journal steht: »Commodore kann seine Kredite nicht zurückzahlen«, dann schreiben das rund um die Welt viele ab. Nicht alle, es gibt da genügend Ausnahmen, die sich der Mühe unterziehen selber zu recherchieren und nachzufragen. Wenn sie aber 1 und 1 zusammenzählen würden, würden sie merken, das kann überhaupt nicht stimmen. Welches Unternehmen kann über Nacht Kredite, die mit den Banken vereinbart sind, zurückzahlen. Das geht überhaupt nicht. Wirklich, 85 Prozent der Unternehmen wären über Nacht zahlungsunfähig.

Wenn es Commodore so schlecht ginge, wäre Commodore mit Sicherheit unter Chapter 11 gegangen, um den Forderungen der Gläubiger zu entgehen, aber das ist nicht der Fall, war



Interviewpartner Harald Speyer, Vice President Europe Commodore International

nicht der Fall und wird nicht der Fall sein.

64'er: Wie sieht denn nun die finanzielle Situation von Commodore tatsächlich aus?

Speyer: Wir haben im ersten Geschäftsquartal 1986, das Geschäftsjahr 1986 von Commodore geht vom 1. Juli 1985 bis 31. Juni 1986, noch einen Umsatzrückgang im Vergleich zum Vorjahr gehabt, im zweiten Quartal (vom 1.10.85 bis 31.12.85; Anm. d. Red.) eine Umsatzsteigerung. Wir haben in diesem zweiten Geschäftsquartal, dem Weihnachtsquartal, mehr Computer verkauft, über eine Million weltweit, als im Vergleichsquartal des Vorjahres. Auch das entgegen aller Branchenprognosen, speziell in den USA, die ja behauptet haben, der Heimcomputer sei tot. Wenn der Heimcomputer tot wäre, hätten wir mit Sicherheit nicht in drei Monaten über eine Million Computer verkauft.

Es hat auch eine leichte Umsatzsteigerung im Wert gegeben. Wir haben zwar nur geringfügig steigern können, aber immerhin eine Steigerung, was ja auch den Vorhersagen einiger »Branchen-Gurus« zuwiderläuft.

Noch ein anderer Aspekt: Wir hatten im Weihnachtsquartal 1984 mehr in den Handel hineingeliefert, als der Handel verkaufen konnte, weil der Handel ein wesentlich höheres Weihnachtsgeschäft 1984 erwartet hatte. Er ist daher mit einem ziemlichen Lagerbestand in das Jahr 1985 hineingegangen. Das Phänomen gibt es in diesem Jahr nicht. Der Handel hat sehr konservativ geordert, hat immer nur das bestellt, was tatsächlich verkauft werden konnte. Wir hatten zum Ende des Jahres 1985 keine Lagerbestände. Das spiegelt sich nun in einem um 400 Prozent höheren Auftragseingang im Januar/Februar 1986 im Vergleich zum Vorjahr. Das hat uns angenehm überrascht. Wir gehen davon aus, daß wir in Deutschland, in Europa und weltweit ein besseres Quartalsergebnis haben werden als vor einem Jahr. Das wäre das zweite Quartal in Folge, in dem wir mehr umsetzen als geplant. So wie wir heute den Markt einschätzen, werden wir auch das vierte Geschäftsquartal 1986 mit einem besseren Ergebnis abschließen, so daß wir insgesamt für das Geschäftjahr 1986 rein umsatzmäßig ein besseres Ergebnis zu erzielen glauben. Ertragsmäßig — wenn ich jetzt mal von außergewöhnlichen Aufwendungen für die beiden Werksschließungen, es wurden, um Überkapazitäten abzubauen zwei Werke in England und in Kalifornien geschlossen, absehe — haben wir mit Sicherheit auch ein besseres Ergebnis im laufenden Geschäftsjahr als im Vergleich zum Vorjahr.

Außerdem konnten unsere Lagerbestände, das schließt Rohmaterialien und Halbfertigprodukte ein, von 449 Millionen Dollar auf 225 Millionen Dollar abgebaut werden.

Wir machen mittlerweile den Umsatz nicht mehr nur mit einem Produkt, wie das noch vor zwei Jahren mit dem C 64 der Fall war. Der C 64 macht heute weniger als 35 Prozent des Umsatzes aus, obwohl er sich stückzahlenmäßig nach wie vor auf einem sehr, sehr hohen Niveau bewegt.

64'er: Können Sie genaue Zahlen nennen?

Speyer: Ja. Wir haben 1984 2,5 Millionen, im Jahr 1985 2 Millionen Computer weltweit abgesetzt. Sicherlich ist das ein Rückgang, aber 2 Millionen ist kein Einbruch wie ihn die Branchen-Gurus vorhergesagt haben, sondern ein sehr, sehr hohes Niveau. Das setzt sich unserer Meinung nach in diesem Jahr fort. Wir haben heute weltweit einen Auftragsbestand, der unsere Produktionskapazität für drei Monate auslastet, was wir im Jahr 1985 oder 1984 nicht hatten. Wir sind durchaus der Meinung, daß der C 64 ein wesentlich längeres Leben haben wird, als wir selber erwartet hatten. Das, was andere versucht haben zu erreichen, ist der C 64 geworden — ein Weltstandard.

Aber wie gesagt, der C 64 ist nur noch für 35 Prozent des Umsatzes verantwortlich, während wir den restlichen Umsatz mit Produkten machen, die wir vor zwei Jahren noch gar nicht hatten: dem C 128, der PC-Serie und dem Amiga. Damit sind wir heute erheblich weniger anfällig gegen saisonale Schwankungen oder Konsumentenverhalten wie noch vor zwei Jahren.

64'er: Wieviel trägt der C 128 dazu bei?

Speyer: Wir haben im zweiten Quartal des laufenden Geschäftsjahres 425 000 Stück weltweit verkauft. Man kann also nicht sagen, daß das ein Flop ist, wie das teilweise auch angesprochen wurde. Es hat noch nie ein Computer in seinem Startquartal auch nur annähernd diese Zahl erreicht.

64'er: Wie viele C 128 wurden bis Ende 1985 in Deutschland verkauft?

Speyer: In Deutschland haben wir im zweiten Quartal des laufenden Geschäftsjahres — dem Weihnachtsquartal — etwas mehr als 60 000 Stück verkauft.

64'er: Was erwarten Sie beim C 128 für die ersten beiden Quartale des laufenden Kalenderjahres?

Speyer: Man muß den 128er unterteilen in den C 128 Standalone und in den C 128 D. Den C 128 Standalone rechne ich zu den Consumer-Produkten. Er wird also saisonalen Schwankungen unterworfen sein. Wir werden wahrscheinlich in den beiden verbleibenden Quartalen des laufenden Geschäftsjahres nicht die Stückzahlen des Weihnachtsquartals erreichen können. Mit Sicherheit läuft er auf einem sehr guten Niveau weiter. Wir schätzen, daß wir vom C 128 weltweit zwischen Januar und Juni eine halbe Million verkaufen werden.

Der C 128 D verkauft sich überraschend gut, und zwar in allen Ländern. Er hat nicht diesen saisonalen Rückgang zu verzeichnen. Beispielsweise wurde der C 128 D in Belgien als offizieller Schulcomputer auserwählt. Hier in Deutschland ist das auch schon in zwei Bundesländern passiert. Der C 128 D ist heute von der Leistung her, Stichwort CP/M, in der Lage, einen Kleinstbetrieb DV-mäßig abzudecken. Er ist immerhin bis 512 KByte RAM ausbaubar.

64'er: Wann wird die Speichererweiterung von 128 auf 512 KByte kommen, und was wird sie kosten?

Speyer: Mit Sicherheit noch bis Mitte dieses Jahres. Sie wissen, wie die RAM-Preise heute sind, es wird sicherlich ein Commodore-like-Preis sein.

64'er: Wie wird sich nach Ansicht von Commodore der 8-Bit-Heimcomputermarkt entwickeln?

Speyer: Wir sind der Meinung, daß der 8-Bit-Heimcomputermarkt ein sehr viel längeres Leben haben wird, als wir noch vor einem halben Jahr glaubten. Es gibt mittlerweile Software für den C 64, die es erlaubt mit einem 8-Bit-Heimcomputer ähnli-

Das Abenteuer lockt

In dem 4. Sonderheft der 64'er-Reihe geht es um Adventures, zu deutsch: Abenteuerspiele. Zwei große Bereiche bestimmen den Inhalt dieses Sonderheftes: Der ausführliche Programmier-Kurs und eine Reihe von Super-Adventures zum Abtippen.

Wer selbst programmieren will, bekommt in unserem ausführlichen Programmierkurs jede Menge Informationen von einem Fachmann: Michael Nickles, unter anderem bekannt durch Gordon Saga, hat auf fast 100 Seiten alles zusammengestellt, was man wissen muß, um Adventures zu programmieren. Er beantwortet ausführlich, wie ein Computer deutsche Eingaben versteht oder wie man ganze Sätze decodiert. Natürlich werden auch fertige Routinen vorgestellt, die die ganze Arbeit übernehmen.

Der Kurs beschreibt auch, wie man den Speicherplatz auf Diskette bei langen Adventures optimal nutzt, Texte speichert und verwaltet, und wie man hochinteressante Grafiken in Abenteuerspiele einbaut.

Ein Thema, das auf den ersten Blick nicht viel mit Adventures gemein zu haben scheint, ist »Künstliche Intelligenz«. Wie verwandelt man seinen Computer in eine »Denkmaschine«? Wie schreibt man Adventures, die »denken« können? Diese Fragen werden ebenfalls ausführlich beantwortet.

Ein paar Themen sind auch für alle Programmierer, nicht nur für die Adventure-Fans, interessant: Deutscher Zeichensatz für den Commodore 64, Dateiverwaltung nicht nur für Adventures sowie viele Tips&Tricks.

Damit man auch etwas spielen kann, bevor man sich sein eigenes Abenteuerspiel schreibt, haben wir die besten Adventure-Listings, die von unseren Lesern eingeschickt wurden, in diesem Sonderheft zusammengefaßt: Die Themen reichen von einer deutschen Version eines Hobbit-ähnlichen Spiels, über das Bestehen schwieriger Zauberprüfungen, einem Gefängnisausbruch und einem spannenden Krimi bis hin zur Strandung auf einer einsamen Insel.

Das 64'er-Sonderheft 4/86 »Abenteuerspiele« ist ab März im Zeitschriftenhandel.

ches zu machen, wie mit einem 16-Bit-Office-Computer. Sie haben ja schon darüber berichtet (Ausgabe 3/86, Seite 8; Anm. d. Red.). Wenn Sie das Programm namens »Geos« gesehen haben, werden Sie feststellen, daß dieses Programm zum Beispiel gegenüber GEM keine Wünsche offen läßt.

64'er: Wird es dann möglicherweise auch Speicher-Erweiterungen für den C 64 geben?

Speyer: Das ist durchaus in unserer Planung. Der C 64 wird also auf keinen Fall sterben. Er wird mit Sicherheit im Laufe der nächsten Jahre Wandlungen erfahren. Er bleibt im Grunde ein C 64, der aber bei weitem über die heutigen Möglichkeiten hinausgehen wird.

64'er: Wie sehen die konkreten Ideen bei Commodore dazu aus?

Speyer: Wir haben natürlich sehr konkrete Ideen, die wir im vorgesehenen Zeitraum verwirklichen werden. Es wäre heute verfrüht, über Details zu sprechen. Sie können sich vorstellen, daß wir im Prinzip mit unserer Planung immer ein bis zwei Jahre vorausdenken. Insofern haben wir natürlich eine ganze Menge Ideen, was wir mit dem C 64 machen werden.

64'er: Man kann also vermuten, daß da in der Zukunft noch einiges auf uns zukommt?

Speyer: Das überlasse ich Ihnen, die Vermutung. Sie müssen sich vorstellen, den C 64, mit einer Verbreitung von rund 6 Millionen Stück, den wirft man nicht einfach weg und sagt: Bitte hier ist etwas Neues. Der wird noch die Fantasie von vielen nachwachsenden Jugendlichen anregen, was man noch damit machen kann. Denn die lernen ja schon das alles kennen, was heute alles existent ist. Und da gibt es immer einen Prozentsatz der Anwender, deren Ehrgeiz es ist, noch mehr damit zu machen. Und so ist auch das Geos-Programm entstanden. Man hat also dem C 64 im Prinzip ein neues Betriebssystem verpaßt. Man hat ihn mit einem Turbo-Lader ausgestattet, der wesentlich schneller ist als alles, was auf dem Markt ist, und das alles softwaremäßig. Insofern können Sie sich die Richtung ausmalen, in die der C 64 gehen wird.

64'er: Sie sind also der Meinung, daß der 8-Bit-Markt speziell mit dem C 64/C 128 weiterleben wird?

Speyer: Ja. Man muß sagen einerseits zur Freude unserer Vertriebsleute, da sich dieses Gerät nach wie vor uneingeschränkter Beliebtheit erfreut. Andererseits würden natürlich unsere Ingenieure liebend gern etwas Neues anstelle des C 64 bringen. Aber noch entscheidet der Markt, was er will, nicht ausschließlich die technologischen Möglichkeiten. Wir können nur Neues anbieten und abwarten, ob der Markt das in der Form will, oder ob er das andere bestehende, verbesserte Produkt weiter akzeptiert. In dem Stadium sind wir heute. Wir sehen, daß der C 64 selbst in Nordamerika heute wieder besser läuft als 1984 im Weihnachtsquartal. Wir haben 1985 im Weihnachtsquartal mehr C 64 verkauft als im Vorjahr.

Insofern muß man sagen, der 8-Bit-Markt ist weder tot, noch befindet er sich in der Agonie (Todeskampf; Anm. d. Red.), so daß man sein Ende voraussehen könnte. Im Gegenteil, wenn man sieht, was an neuen Möglichkeiten rund um den C 64 entsteht, muß man sagen, das Produkt wird sicherlich noch ein extrem langes Leben haben.

64'er: Wird sich dadurch beim Preis des C 64 etwas ändern?

Speyer: Ich glaube nicht, daß es weltbewegende Preisänderungen geben wird. Die Bauteile sind heute auf einem Preisniveau, wo man kaum vermuten kann, daß sie noch weiter in den Keller gehen werden. Es macht absolut keinen Sinn, ein Unternehmen auszuquetschen und es schließlich in den Ruin zu treiben. Jeder muß bei dem Geschäft verdienen, der Handel, aber der Hersteller auch. Schließlich ist es auch im Interesse des Verbrauchers, neue Produkte entwickeln zu können und bestehende Produkte permanent zu verbessern. Nicht der allergünstigste Preis ist entscheidend, sondern die Gewißheit, daß der Käufer auf Jahre hinaus einen solventen Partner hat.

64'er: Das heißt auch, daß mit dem Ausscheiden von Jack Tramiel die Niedrigpreispolitik und -philosophie von Commodore gewichen ist?

Speyer: Wir sind nach wie vor bestrebt, zu einem gegebenen Preis das beste Leistungsverhältnis zu bieten. Beides zu verlangen, den niedrigsten Preis und die beste Leistung, ist gegen das ökonomische Prinzip gerichtet. Man kann nur eine Größe variabel halten. Wir glauben heute, daß der Markt in den meisten Ländern sehr preissensitiv ist, aber das ist nicht das ausschließliche Kriterium. Es existiert durchaus auch die Nachfrage nach Leistung und Leistungsverbesserung.

64'er: Wie äußert sich diese?

Speyer: In Weiterentwicklungen, in neuen Peripherie-Produkten, die kommen werden.

64'er: Auch in der Reparatursituation?

Speyer: Da muß ich sagen, es ist jeder seines eigenes Glückes Schmied. Wenn ich den Computer beim Discounter kaufe, zum absolut günstigsten Preis, darf ich mich nicht wundern, wenn ich den schlechtesten Service in Kauf nehmen muß. Was nicht heißt, daß es da nicht auch Ausnahmen geben kann. Es wird hier in der Computerindustrie etwas versucht, was man mit keinem anderen Produkt macht. Ich glaube nicht, daß ein Konsument, dessen Fernseher entzweigeht, nun den Hersteller des Fernsehers anruft oder ihn beschimpft, sondern sagt: Bei dem Händler kaufe ich nicht mehr. Genau das gleiche gilt auch für Computer. Man sollte sich seinen Handelspartner genau ansehen. Will ich zum günstigsten Preis kaufen und verzichte auf Unterstützung im Reparaturfall, na bitte. Es würde nichts schaden, wenn der Konsument, bevor er kauft, auch mal die Frage nach Service-Möglichkeiten stellen würde. Was passiert, wenn mein Gerät defekt ist? Wie lange muß ich warten? Das ist ein Kriterium, das in der Computerbranche bisher von seiten der Konsumenten völlig außer acht gelassen wurde. Wir haben mit unseren offiziellen Handelspartnern Service-Vereinbarungen von Anfang an getroffen. Diese Handelspartner bieten den technischen Kundendienst und auch Beratung.

Es gibt aber auch genügend Händler, die nicht unsere offiziellen Händler sind, sondern sich größtenteils auf dem grauen Markt eindecken. Für die können wir natürlich überhaupt keine Gewährleistung übernehmen. Hier sollte der Konsument beim Kauf etwas kritischer sein.

64'er: Commodore hält zur Zeit auf dem deutschen Heimcomputermarkt einen Anteil von etwa 65 Prozent ...

Speyer: ... Er lag im Oktober/November bei 68 Prozent. Was übrigens höher war als zu Beginn des Jahres 1985. Wir haben also im letzten Jahr Marktanteile hinzugewonnen.

64'er: Worauf führen Sie das zurück?

Speyer: Nun, das führe ich in erster Linie darauf zurück, daß die meisten Anbieter, die 1984 noch im Markt waren, heute keine Rolle mehr spielen. Speziell die britischen Anbieter sind heute weltweit, mit Ausnahme in Großbritannien, praktisch nicht mehr existent.

64'er: Mit welchen Maßnahmen wollen Sie diesen Marktanteil in Zukunft sichern?

Speyer: Wir haben eine ganze Bandbreite von Maßnahmen. Wir haben heute zwei Produkte, wo wir früher eins hatten. Ich habe schon eingangs gesagt, daß der C 64 ein lebendes Produkt ist. Es wird sich weiter entwickeln. Und letztlich glauben wir, daß der Preis nicht das alleinige Kriterium ist. Der C 64 war nie das billigste Gerät gewesen, das muß einmal ganz deutlich herausgestellt werden. Es gab immer preisgünstigere Geräte im Markt, und trotzdem hat sich der C 64 durchgesetzt, einfach, weil das Preis-/Leistungsverhältnis stimmte. Heute kommt natürlich das nicht mehr überschaubare Angebot an erhältlicher, auch kommerzieller Software der unterschiedlichsten Richtungen dazu. Es gibt eine Vielfalt von Software, die wirklich nicht mehr zählbar ist. Zum Einstieg für die Jugend gibt es natürlich sehr, sehr viele Programme, und — wir dürfen uns da kein X für ein U vormachen — jeder, der sich einen C 64 kauft, hat von seinen Freunden ein riesiges Reservoir an kopierter Software. Da ist auch aus unserer Sicht nichts dagegen einzuwenden, solange das nicht professionell kopiert wird. Wenn das zur Weitergabe unter Freunden ohne Entgelt bestimmt ist, so würde ich sagen, ist dagegen absolut nichts einzuwenden. Ich spreche da natürlich nur in unserem Namen; ich kann nicht für andere Software-Anbieter sprechen, die vielleicht etwas dagegen haben.

64'er: Wie sieht Ihrer Meinung nach der Heimcomputer in 2 und in 5 Jahren aus?

Speyer: Wie er in zwei Jahren aussehen wird, läßt sich im großen und ganzen heute schon absehen. Der Markt ist nicht so technologieabhängig, wie es manchmal versucht wird zu suggerieren. Der Markt ist wesentlich konstanter, auch im Computerbereich. Ich glaube nicht, daß sich innerhalb von zwei Jahren der Heimcomputer drastisch verändern wird. Er wird leistungsfähiger, gar keine Frage.

In fünf Jahren, bin ich der Meinung, wird Sprachein- und -ausgabe eine Rolle spielen. Das dürfte im Prinzip die wesentlichste Neuerung sein. Das ist ja heute schon existent, aber viel zu teuer. Es ist nur eine Frage der Zeit, bis die erforderliche Speicherleistung so kostengünstig wird, um Sprachein- und -ausgabe auch in ein Consumerprodukt einzubauen.

Die Grafikmöglichkeiten werden, das haben wir mit dem Amiga vorgezeichnet, im Heimcomputerbereich bei weitem über das hinausgehen, was wir heute haben. Ein Amiga kommt immerhin an die Möglichkeiten heutiger großer CAD-Systeme heran. Andere Features sind die Synthesizer-Möglichkeiten, die es durchaus mit professionellen Synthesizern aufnehmen können.

Das heißt der Heimcomputer wird ein Rundum-Gerät; er ist nicht mehr der Rechner, von

dem er seinen Namen ableitet, sondern er wird das Kommunikationszentrum der Zukunft werden.

64'er: Wird man von Commodore auch ein derartiges Kommunikationszentrum erwarten können?

Speyer: Mit Sicherheit.

64'er: Zu welchen Preisen?

Speyer: Es ist sehr schwer über Preise zu spekulieren, die in die Zukunft gerichtet sind. Wir wissen alle nicht, wie sich Zinsen und Löhne entwickeln, wie sich Steuern und Zölle entwickeln. Wir können einigermaßen abschätzen, wie sich die Technologie und die Preise entwickeln werden. Wir brauchen dazu aber nicht nur neue Bauteile, die erst entwickelt werden müssen, sondern auch neue Produktionstechnologien. Wenn wir alle Faktoren zusammen haben, kann man abschätzen, ob es zu heutigen Preisen möglich sein wird. Aber es spielen so viele Faktoren mit.

64'er: Der Amiga scheint der Hoffnungsträger für Commodore geworden zu sein. Wie sehen Sie den Amiga?

Speyer: Der Begriff Hoffnungsträger ist hochgeputscht worden. Das Produkt Amiga ist in Europa für dieses Geschäftsjahr mit ganzen drei Prozent vom Umsatz geplant. Für Nordamerika wird der Amiga, obwohl schon seit September lieferbar, mit weniger als zehn Prozent des Umsatzes geplant. Da kann uns keiner unterstellen, daß unser ganzes Hoffen, unser ganzes Wohl und Wehe von diesem Produkt abhängt. Das ist schlicht Blödsinn!

Wir wissen, daß der Amiga seinen Weg machen wird. Wir wissen aber auch, daß der Amiga erst einmal seinen Markt finden muß.

64'er: Wo sehen Sie den Amiga plaziert?

Speyer: Er ist zunächst einmal ein Personal Computer für Anwender, die viel im Grafikbereich zu tun haben. Er ist ein Computer für die vertikalen Märkte (spezielle Branchenlösungen für Musiker, Grafiker, Ingenieure oder Designer; Anm. d. Red.). Es entsteht aber auch zu unserer Freude viel Software für den horizontalen Markt (Standardsoftware wie Textverarbeitungs- oder Datenbankprogramme für den allgemeinen, branchenübergreifenden Einsatz; Anm. d. Red.). Damit meine ich professionelle, horizontale Software, die den Einsatz im allgemeinen Bürobetrieb erlaubt und die sicherlich auch von den zusätzlichen Möglichkeiten, die ein Amiga gegenüber den bisherigen PCs bietet, Gebrauch macht. Dann liegt es beim Anwender zu entscheiden, ob er das leistungsfähigere System nutzt oder ein bereits bekanntes System, das ja, wenn ich von MS-DOS spreche, auch schon fünf Jahre alt ist.

64'er: Und trotzdem wird es für den Amiga einen MS-DOS-Emulator geben?

Speyer: Es wird ihn geben. Es gibt durchaus Anwender, die sagen, ich möchte den Amiga nutzen, aber ich möchte die Programme, die ich bisher eingesetzt habe, weiterhin verwenden. Diese Möglichkeit wollen wir schaffen. Dann ist er aber auch nicht mehr als ein PC und nicht schneller als ein PC heute ist.

Wir haben noch zu keinem unserer Produkte, speziell hier in Europa und in Deutschland, auch nur annähernd anfangs eine solche Unterstützung von der Software-Welt erlebt.

64'er: Es war bei der Vorstellung des Amiga in New York die Rede davon, daß es den Amiga in verschiedenen Versionen geben wird. Ist das immer noch so?

Speyer: Nicht diesen Amiga. Es wird mehr Produkte geben, die die Grafik- und Synthesizerleistung des Amiga, das heißt, die drei Custom-Chips, haben werden. Das sind nicht unbedingt die gleichen Geräte. Es wird Geräte geben, die über die heutige Leistung des Amiga hinausgehen werden, die durchaus mehr auf den technisch/wissenschaftlichen Bereich zielen. Damit soll im CAD/CAM-Bereich eine kostengünstige Möglichkeit geschaffen werden für die vielen Klein- und Mittelbetriebe, die heute noch gar nicht daran denken, derartige Systeme einzusetzen.

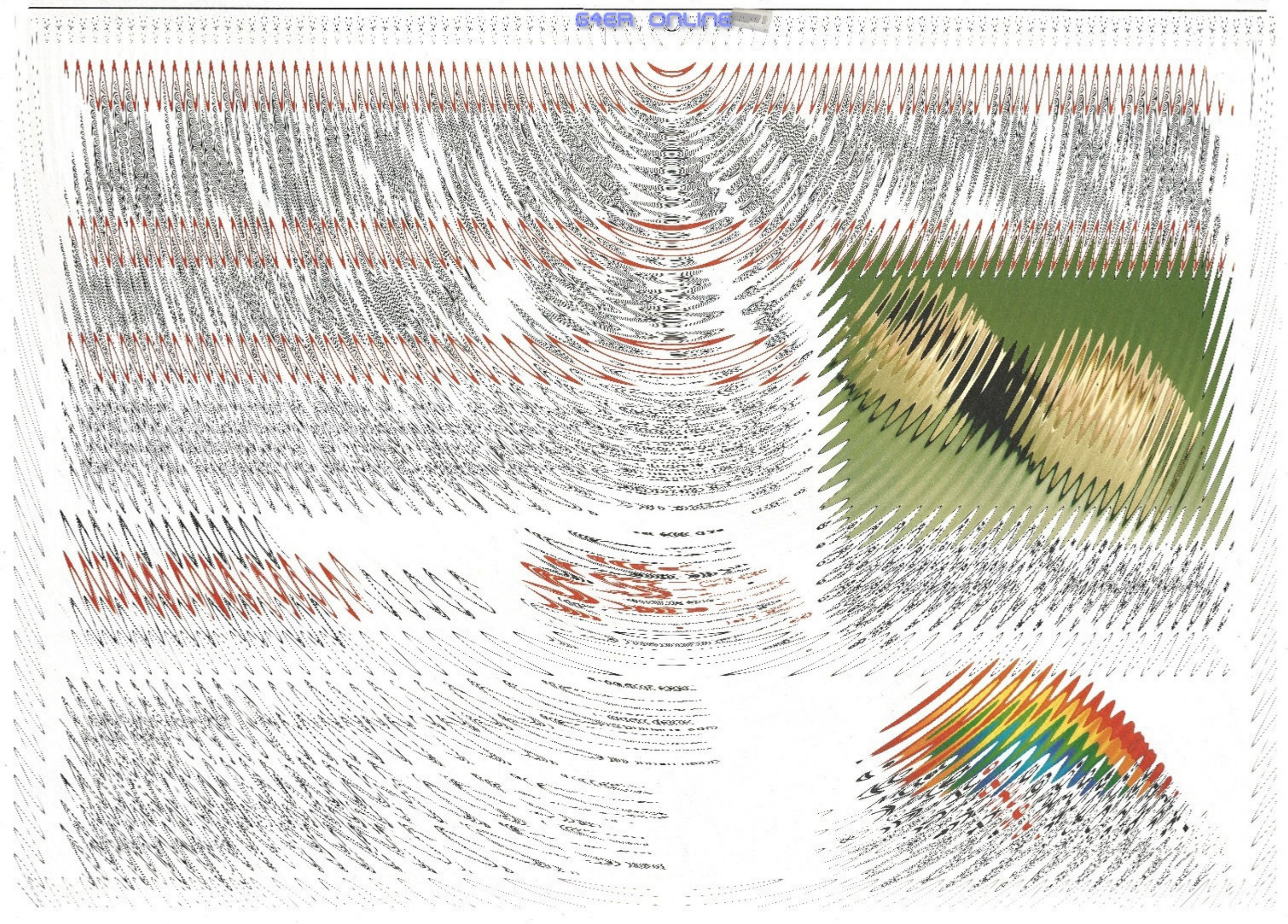
Wir denken auch darüber nach, die Amiga-Chips eines Tages im Heimbereich zu verwenden.

64'er: Was kann man sonst noch an neuer Peripherie von Commodore erwarten?

Speyer: Wir denken zur Zeit über längerfristige Konzepte für Peripheriegeräte nach. Wir werden sicherlich künftig parallel zu den existierenden Floppies auch leistungsfähigere anbieten, die dann natürlich auch mehr Geld kosten. Wir sind mitten im Entscheidungsprozeß für eine komplette Druckerpalette, die vom Low-Cost-Drucker über Farbdrucker für den Amiga bis zum Laserprinter reicht.

64'er: Man wird also von Commodore in Zukunft noch einiges zu erwarten haben. Herr Speyer, ich danke Ihnen für dieses Gespräch.

(aa)



DFÜ-NEWS

INTERPOOL

Es ist nun schon fast zwei Jahre her, daß die erste private Mailbox in Deutschland an das öffentliche Telefonnetz angeschlossen wurde. Damals bestand der Benutzerkreis aus wenigen Hardwarefreaks, die sich ihre Akustikkoppler oder Modems zumeist selbst gebastelt hatten. Doch die »Szene« hat sich gewandelt. Baupläne und Basteltips für Modems sind out, spätestens seitdem komplette Koppler auch für Schülergeldbeutel erschwinglich sind. Und der Trend hält an. Immer mehr Leute wollen sich des neuen Mediums bedienen. Mailboxen schießen wie Pilze aus dem Boden. Mittlerweile gibt es bundesweit an die 300 elektronische Briefkastensysteme. Eine beachtliche Entwicklung, kaum ein anderes Medium hat so schnell Fuß gefaßt.

Damit die Telefongebühren jedoch im Rahmen bleiben, bewegen sich die meisten Mailboxbenutzer vorwiegend im Nahgesprächsbereich. Datex-P, das Datennetz der Deutschen Bundespost, ist für den Großteil auch zu teuer, zumal kaum eine Datex-Box gebührenfrei zu benutzen ist. Als unmittelbare Folge daraus hat ein beträchtlicher Anteil der Boxen regionalen Charakter, ein Erfahrungsaustausch mit DFÜ-Gruppen aus anderen Bundesländern ist kaum möglich. Bisher kümmerte sich jeder Sysop stets nur um die eigene Box. Der Blick über den »Tellerrand« hinaus fehlte.

Genau das war ein Thema beim Kongreß des Hamburger Chaos Computer Clubs, der Anfang März stattfand.

Dort setzten sich etwa 20 Mailboxbetreiber an einen Tisch und faßten einen interessanten Beschluß: Es soll der Verein »Interpool« gegründet werden. In ihm wollen sich die Boxbetreiber zu einem Informationsaustausch zusammenschließen. Dieser Austausch kann natürlich nur über einen zentralen Sammler, dem Pool eben, ablaufen. Hier soll jeder Boxbetreiber die interessanten News aus seiner Box einspielen. Im Gegenzug bekommt er für seine Box wiederum die interessantesten Mails der anderen bereitgestellt. So sollen die zunächst nur regional verfügbaren Nachrichten schnell bundesweit verteilt werden. Zu einem späteren Zeitpunkt ist sogar eine Zusammenarbeit über Deutschlands Grenzen hinaus geplant. Durch den Interpool sollen die Mailboxen zu einem wirklich potenten neuen Kommunikationsmedium werden. Der Vereinszweck »Völkerverständigung« ist somit kaum zu hoch gegriffen. Natürlich gibt der Verein seinen Mitgliedern einige Auflagen vor. So sollen nur Boxen, die zuverlässig auch über einen längeren Zeitraum arbeiten, aufgenommen werden. Denn die Anzahl derer, die die Lust an ihrer Box nach einiger Zeit verlieren, ist beträchtlich. Auch die juristische Seite muß geklärt werden. Eine Art »elektronisches Copyright« soll eingeführt werden. Demnach dürfen die Mails aus den einzelnen Boxen nur in anderen aufgenommen werden, wenn diese auch dem Interpool angehören. Die Verbreitung darüber hinaus soll untersagt werden. Die Urheberrechte sollen jedoch beim Autor bleiben. Da der Verein die Gemeinnützigkeit anstrebt, können nur private und kostenfreie Boxen aufgenommen werden. Kommerzielle oder Werbezwecken dienende Systeme sind deshalb ausgeschlossen.

Im Moment suchen die Initiatoren des Vereins noch nach einer möglichst preiswerten Realisierung des eigentlichen Pools. Der entsprechende Computer soll über Datex-P erreichbar sein. Denn das kommt bei größeren Datenmengen wesentlich günstiger als eine normale Telefonverbindung.

Hier könnten dann die Informationen, nach Themenkreisen sortiert, gesammelt und den Mitgliedern zur Verfügung gestellt werden.

Der Interpool ist wohl der erste bundesweite Versuch, ein alternatives Datennetz quasi »von unten« her aufzubauen. Eine Zensur wie in anderen Medien ist kaum möglich. Der Bildung von Informationsmonopolen könnte so entgegengewirkt werden. Wollen wir hoffen, daß es nicht nur bei der Idee bleibt.

(BHP/hm)

LITERATURTIPS FÜR HACKER UND DFÜ-FREAKS

Aus dem Angebot der inzwischen auf dem Markt befindlichen Bücher für und über Hacker haben wir diesmal ein paar bemerkenswerte Exemplare herausgefischt:

Ammann, Lehnhardt, Die Hacker sind unter uns.

»Die Hacker sind unter uns« ist kein Lehrbuch für Hacker.

Es ist vielmehr eine gut und interessant geschriebene Einführung in die amerikanische Hackerszene. Aber auch deutsche Hacker kommem zu Wort. Berichte über die Aktivitäten deutscher Hacker in nationalen und internationalen Netzwerken und ein kleiner technischer Anhang schließen das Buch ab.

Fazit: Ein gut recherchiertes und flüssig geschriebenes Buch, das man auch mal dem Freund, der Freundin oder den Eltern in die Hand drücken kann, wenn sie immerzu wissen wollen, was man denn die ganze Zeit mit seinem Computer und dem Telefon treibt.

Info: Ammann, Lehnhardt, Die Hacker sind unter uns, Heyne Verlag München, 1985, ISBN 3-453-47055-9, Preis: 9,80 Mark.

Fernmeldetechnisches Zentralamt, Datex-P-Handbuch.

Das Original, direkt von der Post. Das umfassendste, was über Datex-P derzeit erhältlich ist: Ein DIN-A4-Ordner voll mit knochentrockener Technik über Datex-P. Vieles davon ist für den praktischen »Hackbetrieb« aber völlig unnötig. Das, was man als Hacker über Datex-P wissen muß, steht zum Teil auch in der Kurzbedienungsanleitung zu Datex-P, die es bei der Post kostenlos gibt.

Info: Fernmeldetechnisches Zentralamt, Datex-P-Handbuch, 1983, Fernmeldetechnisches Zentralamt Darmstadt. Bestellnummer: KNr 652 430 200-5, Preis 50 Mark.

Günter Wahl, Minispione V.

Die Reihe Minispione, die seit Erscheinen des ersten Bandes im Jahre 1969 zu den absoluten Rennern der Top-Buchreihe gehört, berichtet im Band 174/175 (der fünften Ausgabe der Minispion-Bände) auch einiges über Zusätze und Schaltungen rund ums Telefon. »Phone-Freaks« kann dieses Buch wärmstens empfohlen werden. Es werden, neben Grundlagen über das Telefonnetz, auch Schaltungen rund ums Telefon, verbunden mit Tips für »weitergehende Nutzung«, geboten.

Info: Günter Wahl, Minispione V, Frech-Verlag Stuttgart, 1980, ISBN 3-7724-0474-X, Preis: 18,80 Mark.

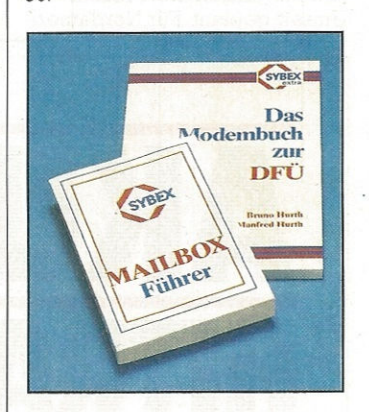
Deutsche Bundespost, Amtliches Verzeichnis der DATEX-L und DATEX-P-Teilnehmer der Deutschen Bundespost.

An sich das, was der angehende »Hacker« braucht — ein vollständiges Verzeichnis aller Datex-P-Teilnehmer. Praktisch so etwas wie das Telefonbuch für den Datex-Rechnerverbund. Doch bereits beim ersten Durchblättern kehrt schnell Ernüchterung ein: Da sind doch fast alle NUAs vergessen worden! Und so sinkt der Gebrauchswert dieser Broschüre nahezu auf Null, wenn man nach heftigem Blättern feststellen muß, daß bestenfalls nur bei 0,5% der Einträge die Datex-Nummern vorhanden sind. Geschehen ist das Ganze wohl aus Angst vor diesen ominösen Hackern, denn in der Nullausgabe dieses Verzeichnisses waren noch alle NUAs vorhanden. Wer's trotzdem will:

Info: Deutsche Bundespost, Amtliches Verzeichnis der DATEX-L und DATEX-P-Teilnehmer, Bestellungen über das Fernmeldeamt 2, Postfach 10 00 12, 8500 Nürnberg 1 oder Müller Adress GmbH, Pretzfelder Str. 7—11, 8500 Nürnberg 90, Preis: 12 Mark.

Unterrichtsblätter der Deutschen Bundespost, Ausgabe B Fernmeldewesen.

Technisch knallharte Informationen (Anschaltung automatisierter Dienste an das Fernsprechnetz etc.) wechseln sich ab mit langatmig Uninteressantem (Werkstoffe, die man Kunststoffe nennt). Für technisch Bewanderte (und welcher Hacker ist das nicht) ein absolutes Muß, zumal der Preis lächerlich gering ist: Ein Abonnement für ein ganzes Jahr (12 Ausgaben) kostet nur DM 7,20! Bestellungen erfolgen mit dem Formular »Zeitungsbestellung«. Erhältlich bei: Schriftleitung der Unterrichtsblätter der DBP, Fernmeldewesen, Postfach 555, 2000 Hamburg 36.



Hurth, Bruno, Sybex Mailboxführer

Dieses Buch richtet sich an alle DFÜ-Neulinge, die den ersten Kontakt zu diesem Kommunikationsmedium suchen. Im Mailboxführer werden nämlich die bekanntesten Mailboxen Deutschlands vorgestellt. Dabei wird beschrieben, wie man in sie hineinkommt und mit welchen Einstellungen sie arbeiten. Auszüge (Kommunikationsprotokolle) aus den Mailboxen geben zum Teil schon einen guten Eindruck darüber, was in den Boxen geboten wird. Außerdem erfährt man, wie man sich bei den verschiedenen Mailboxen eine private Box einrichten lassen kann und was allgemein beim Umgang mit Mailboxen beachtet werden sollte.

Info: Bruno Hurth, Sybex Mailboxführer, Sybex Verlag Düsseldorf, 1985, ISBN 3-88745-620-3, Preis: 14,80 Mark.

Hurth, Bruno und Manfred, Das Modembuch zur DFÜ

Ein Buch für den Hacker und den Datex-P-Anfänger. Hier findet man vieles, was man häufig vergebens sucht: Adressen von Mailboxen und ein nach Sachgebieten sortiertes Verzeichnis von Datenbanken und deren Hosts. Die erwähnten Datenbanken sind alle dem Euronet DIANE angeschlossen. Ebenso gibt es ein nach Orten angelegtes Verzeichnis von Informationsvermittlungsstellen in Deutschland.

Das Datenbank- und Hostverzeichnis nimmt etwa die Hälfte des Buches ein. In der anderen Hälfte erfährt man Wissenswertes über alles, was mit DFÜ zu tun hat: Schnittstellennorm V.24, Akustikkoppler, Btx, Datex-L und Datex-P.

»Das Modembuch zur DFÜ« ist ein gelungenes Nachschlagewerk für alle die, die sich mit professioneller Datenkommunikation beschäftigen. Aber auch derjenige, der seinen ersten Kontakt zu professionellen Datennetzen sucht, wird seine Freude an diesem Sybex-Buch haben. (BHP/hm)

Info: Manfred und Bruno Hurth, Sybex Verlag Düsseldorf, 1985, ISBN 3-88745-619-X, Preis: 24,80 Mark.

NEUER DATAPHON-KOPPLER

Woerltronik bietet ab Mitte April einen neuen Typ des bekannten Akustikkopplers Dataphon an. Der neue Dataphon s 21/23 Kombi soll gegenüber dem s21d einige Vorteile haben: Induktive und akustische Kopplung, Btx-fähig (mit DBT 03-kompatibler Rundbuchse) Übertragungsgeschwindigkeiten 300, 600, 1200 bit/s im Hauptkanal und 75 bit/s im Nebenkanal. Bei Halbduplex sind auch 1200/1200 bit/s möglich (Bei Datex-P ist allerdings Vollduplex nötig). Über die RTS-Leitung soll eine automatische Sende- und Empfangsumschaltung realisiert werden können. Ein automatischer Geschwindigkeitswandler schaltet von selbst von 1200 bit/s auf 75 bit/s im Rückkanal um. Die Stromversorgung erfolgt über Batterie, Netzteil oder V.24-Schnittstelle.

Der »Universalkoppler« soll etwa 400 Mark kosten. (hm)

Info: Wörlein GmbH, Hindenburgstr. 37, Postfach 4, 8501 Cadolzburg, Tel. 09103/8294 oder 8552

MERLINKARTE NOCHMALS VERBESSERT

In der letzten Ausgabe stellten wir Ihnen die Merlin 4-fach-Expansion-Port-Karte vor. Zusammen mit dieser bietet Merlin nun einen verbesserten Modulgenerator an. In einem EPROM können nun bis zu acht Programme untergebracht werden. Jedem Programm kann ein Name gegeben werden. Wird eine solches EPROM am Expansion-Port angeschlossen, erscheint nach dem Einschalten des C 64 oder C 128 ein Menü auf dem Bildschirm, über das eines der Programme gewählt werden kann. Jedes Programm selbst kann aus acht Teilen bestehen. Beispielsweise einem Basic-Programm, das sieben Maschinenroutinen an verschiedenen Adressen verwendet. Beim Initialisieren des Moduls werden alle Programmteile an die richtige Adresse verschoben. Während ein Modul zusammengestellt wird, zeigt ein Byte-Zähler ständig an, welche Speicherkapazität das EPROM haben muß. (hm)

ALS GESCHENKIDEE ...

für Computerfreunde bietet Döbbelin und Böder eine Disky Archivbox in den Farben Rot, Blau und Gelb an. Nach Wunsch mit zehn farbigen Disketten 5¼ Zoll (einfache Dichte). Ein anderes Set enthält neun schwarze Disketten und ein Katz- und Mausspiel für den C 64 oder besteht aus einer Disky-Archivbox (rot) und je zwei Disky 5¼ Zoll (einfache Dichte) in rot und blau, einem Mini-Reinigungsset für Diskettenlaufwerke und verschiedenen Diskettenaufkleber. (hm)

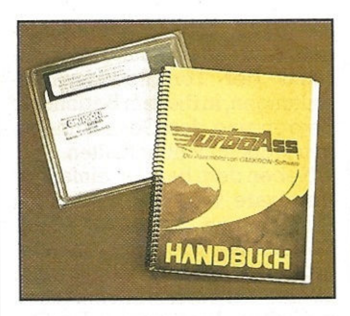
Info: Döbbelin und Böder, Wickerer Str. 50, 6093 Flörsheim am Main, Tel. 06145/5020

BETRIFFT: BESTELLUNGEN UND ANFRAGEN AN DIE B.H.P.

Anschrift:
B.H.P.
Die Bayrische Hackerpost
c/o Basis
Adalbertstr. 41b
8000 München 40
Tel. 089/808447/∅
(Peter Haenelt)

Bestellungen von Freeware-Disketten können nur bei Vorauskasse bearbeitet werden, da ein Versand per Nachnahme zuviel Arbeit und Kosten verursacht. Der entsprechende Betrag sollte in Bargeld oder als Verrechnungsscheck der Bestellung beigelegt werden. Beträge bis DM 5,— können mit Briefmarken (kleine Werte) abgegolten werden.

Eingeschriebene Sendungen können leider nicht angenommen werden und gehen an den Absender zurück.



TURBO-ASS JETZT MIT MACROS

Turbo-Ass, der von den Entwicklern, Omikron Software, als der schnellste Assembler für den C 64 bezeichnet wird, gibt es jetzt in einer erheblich verbesserten Version.

»Turbo-Ass Macro+« bietet auch die Programmierung von Assembler-Macros. Macros dürfen sich selbst oder andere aufrufen. Die Parameter-Übergabe ist sehr komfortabel, es können sogar Op-Codes als Parameter verwendet werden. Damit man Labels mehrfach benutzen kann, sind auch Blockstrukturen kein Problem. Bedingte Assemblierung und Assemblerschleifen sind ebenso möglich, wie variable Label und Einbindung von Unterprogrammen von Diskette. Zu guter Letzt wurden weitere Befehle zur komfortablen Tabellenerzeugung integriert.

Trotz dieser Zusatzfunktionen sind fast alle anderen Eigenschaften des Turbo-Ass erhalten geblieben: Geringer Speicherverbrauch, sehr schnelle Assemblierung, integrierter komfortabler Full-Screen-Editor. Auch der Preis ist derselbe: »Turbo-Ass Macro+« kostet immer noch 139 Mark. Wer schon den alten »Turbo-Ass« hat, kann ihn für 20 Mark gegen die neue Version austauschen. (bs)

Info: Turbo-Ass Macro+, Omikron-Software, Erlachstr. 15, 7534 Birkenfeld, Tel. 07082/5386, 139 Mark

VIDEO-VERTEILER

Bei Schulungen, Messen und Ausstellungen tritt häufiger das Problem auf, einen Computer-Bildschirm mehreren Leuten sichtbar zu machen. Mit geringem Aufwand soll dies mit einem Videoverteiler von GES möglich sein. Das Gerät soll über einen Standard-BAS-Eingang mit 75-Ohm-Impedanz und zehn Ausgänge verfügen. Es sollen zwei bis zehn Monitore angeschlossen werden können. Die eingebaute Elektronik soll den Pegel der Videosignale konstant halten, das heißt, das Ursprungssignal wird nicht überlastet. (hm)

Info: GES, H. Graf, Postfach 1610, 8960 Kempten, Tel. 0831/6211

NEUE SOFTWARE ZUM TROMMELN

Besitzer der »Digital Drums« beziehungsweise der »RSC Drums« können nun mit neuer Software noch mehr aus ihrer Trommelplatine herausholen. »Digital Micro Rhythm« ist ein sehr komfortabler Rhythmus-Editor mit Real-Time-Eingabemöglichkeit.

»Digital Micro Drum« ist ein Programm, das die Ansteuerung der acht Trommelsounds über Drum-Pads oder Midi-Synthesizer und -Sequencer erlaubt.

Zwei Sound-Disketten mit insgesamt 120 neuen, digitalisierten Sounds sorgen für eine große Trommelvielfalt. Speziell für Midi-Synthesizer in Kombination mit dem C 64 gibt es »Digital Midi Drum«. Über den Synthesizer können 27 verschiedene Sounds angesprochen werden.

Wer in Stereo trommeln möchte: Eine Stereo-Version der Platine und Software namens »Digital Micro Rhythm S+« gibt es ebenfalls. 12 verschiedene Trommelklänge stehen hier zur Verfügung. (bs)

Info: von Quadt Studio, Hamacher/Missing GbR, Von-Quadt-Str. 41, 5000 Köln 80, 0221/681457
Digital Micro Rhythm: 89 Mark
Digital Micro Drum: 89 Mark
beide Programme mit Drum-Platine: 249 Mark
Digital Midi Drum (ohne Platine): 76 Mark
Sound-Disks 1 & 2: je 59 Mark
Digital Micro Rhythm S+: 249 Mark

NEUER »HAND-HELD«-JOYSTICK

Aus England kommt ein Steuerknüppel mit sehr ungewöhnlicher Form. Der Konix-Joystick ist für Zwei-Hand-Betrieb konzipiert: Die linke Hand hält den Joystick und betätigt den Feuerknopf, die rechte steuert. In der Redaktion haben wir den Konix kurz aber intensiv getestet: Der Konix ist sehr präzise wie auch schnell. Die manuell erreichbare Feuergeschwindigkeit ist dafür etwas geringer als bei herkömmlichen Joysticks. Der Konix eignet sich für alle Arten von Spielen, weniger für Zeichenprogramme. Bei längerem Spielen kann es zu Ermüdungen und Krämpfen in der Halte-Hand kommen. Obwohl der Konix sehr zerbrechlich aussieht, ist er ungemein stabil: Mikroschalter und Stahlkern im Griff sorgen für ungetrübtes Spiel-Vergnügen. Unsere Meinung: Ein Joystick, den man in Erwägung ziehen sollten. Sicherheitshalber sollte Sie ihn vor dem Kauf mal in die Hand nehmen, ob er Ihrer Handgröße entspricht. (bs)

Info: Konix-Joystick, Rushware, An der Gümpgesbrücke 24, 4044 Kaarst 2, etwa 60 Mark



LEHRERFORTBILDUNG

Die Akademie für Lehrerfortbildung in Dillingen sorgt für die Aus- und Fortbildung im Bereich »Computertechnik im Unterricht«.

Momentan werden Computer im Unterricht schwerpunktmäßig in den Fächern Informatik und Technik eingesetzt. Der Computer bietet aber auch in Biologie, Physik und Chemie fantastische Möglichkeiten. Diese Erkenntnis und die ständig steigende Nachfrage nach Fortbildungskursen für Computertechnik, veranlaßte die Akademie für Lehrerfortbildung die Seminarangebote zu erhöhen.

Physikalische Experimente sollen mit Computerunterstützung eindrucksvoller vorgeführt werden und der Computer soll auch die Durchführung von Experimenten gestatten, die bisher im Unterricht nicht durchgeführt werden konnten.

Solche Probleme will die Akademie für Lehrerfortbildung in Dillingen lösen. Der Akademiebericht (Bild) »Computerpraxis bei naturwissenschaftlichen Experimenten« und die Fortbildungsseminare sollen interessierte Lehrer in die Lage versetzen einen Computer zu programmieren (CBM 4000, 8000 oder C 64) und Schnittstellen sinnvoll einzusetzen.

Der Akademiebericht soll keinesfalls das Studium von Gerätehandbüchern oder Lehrmaterialien zur Informatik ersetzen. Das Ziel des Berichts war zu zeigen, wie mit geringem finanziellen und technischem Aufwand der Computer als Hilfsmittel im Unterricht eingesetzt werden kann.

Anleitungen zum Selbstbau von Zusatzgeräten nehmen einen relativ breiten Raum ein. Die dabei vorgeschlagenen Schaltungen sind nicht exklusiv für »Bastler« gedacht, sondern sollen vor allem dem Verständnis von Meß- und Steuervorgängen mit Computerhilfe dienen. Allerdings wendet sich dieser Akademiebericht hauptsächlich an solche Leser, für die Tastatur und Bildschirm nicht die einzigen Kommunikationsmittel zum Mikrocomputer sind. Methodische Folgerungen und die Diskussion von weiteren Einsatzmöglichkeiten von Computeranlagen sind zwar häufig Gegenstand der Fortbildungsveranstaltungen, in diesem Bericht zur Begrenzung des Gesamtumfanges jedoch nicht enthalten.

Neben dem Bau von einfacher Hardware wird eine Reihe von Programmen hierzu erläutert. Dabei kam es den Autoren weniger auf Programmoptimierung, als auf das Verständlichmachen ihres Grundgerüstes an. Der Bericht bringt ferner auch die Grundlagen des Aufbaus und der Bedienung einer Mikrocomputeranlage, soweit sie zum Verständnis und zur Durchführung der Experimente, Simulationen und der sonstigen Programme nötig sind.

Dieser 240 Seiten starke Akademiebericht ist nicht nur für Lehrer sehr interessant. Er kann gegen Rechnung oder Scheck bestellt werden und kostet 17 Mark. (do)

Info: Akademie für Lehrerfortbildung, Kardinal-von Waldburg-Straße 6, 8880 Dillingen a. d. Donau, Tel. 09071/2030

NEUER VIZA-DISTRIBUTOR

Der Exklusivvertrieb der gesamten Viza-Produktpalette für den deutschen Sprachraum wurde von Data Technology Management (DTM) in Wiesbaden übernommen. Wie der Viza-Hersteller Microtron mitteilte, wurde die abgelaufene Distributorvereinbarung mit der Münchener Firma Interface Age nicht verlängert. Für DTM habe man sich aufgrund guter Erfahrungen mit dieser Firma im Software-Service der Viza-Produkte Vizawrite und Vizastar im letzten Jahr entschieden.

(Detlev Hoppenrath/hm)

NEUE PROZESSOR-GENERATION

Schnellen Datentransfer zwischen RAMs, I/O-Bausteinen und CPU erlaubt der neu von ProzessLogic entwickelte Chip 72017. Das ist ein 16-Bit-Mikroprozessor, der nach dem Packed-Data-Transfer-Prinzip arbeitet. Diese Art des Datenverkehrs zwischen Speicher und CPU erlaubt eine bis zu 40 Prozent höhere Arbeitsgeschwindigkeit des Prozessors gegenüber normalen 16-Bit-Typen wie dem 68000.

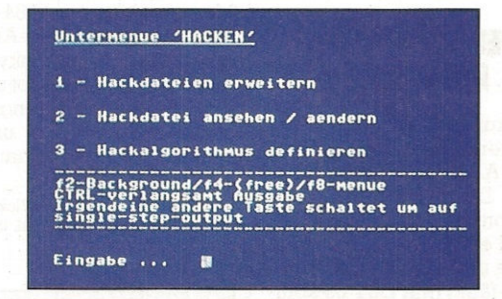
Kurz die Funktion: Normalerweise wird die Adresse auf den Bus gelegt und dann über die 16 Datenleitungen ein 16-Bit-Datum (Word) in die selektierte Speicherzelle geschrieben oder von dort gelesen. Selbst wenn für einen Code nur fünf 1-Bits gebraucht werden, müssen alle Datenleitungen angesteuert werden, auch die, die nur 0-Bits übertragen. Deutlich anders funktioniert der 72017. Durch eine ausgeklügelte Schaltung mit einem raffinierten Timing (MMU 72022) ist es ProzessLogic gelungen, den Adreßbus so zu multiplexen, daß über die eigentlich brachliegenden Datenleitungen schon das nächste Word, zumindest teilweise, gelesen werden kann. Werden für zwei aufeinanderfolgende 16-Bit-Words nur je acht 1-Bits benötigt, können mit einem Datenzugriff beide gleichzeitig geschrieben oder gelesen werden.

Besonders interessant ist die Stackverarbeitung des 72017, der nach dem FINO-Prinzip (First in, Never out) arbeitet. Hier ist der Progammierer endlich nicht mehr gezwungen vor einer RTS-Anweisung eventuelle PHP-Befehle rückgängig zu machen. Das macht der 72017 automatisch. Außerdem wird so eine beliebige Schachtelung von Unterprogrammebenen ohne Stack-Überlauf möglich.

Würde Commodore den 68000 des Amiga durch einen 72017 ersetzen, könnte man eine durchschnittliche Geschwindigkeitserhöhung von 30 Prozent feststellen. Diskettenzugriffe würden sogar um durchschnittlich 60 Prozent beschleunigt.

Außerdem arbeitet der 72017 mit den neuen Assoziativ-Speichern (AAM = »Associative Access Memory«) zusammen. Diese Speicher sind ein neuer Schritt vorwärts in der Speicher-Technik. Im Gegensatz zu normalen RAMs, die bei Angabe einer Adresse den Inhalt auf den Datenbus legen, gibt ein AAM bei Angabe eines Inhalts die dazugehörige Adresse auf den Adreßbus. AAM-Chips gibt es augenblicklich schon in einer 2764-pinkompatiblen Version mit der Bezeichnung 27AR64 und 27AW64. Die AR-Version (Associative Read) verhält sich bei der Programmierung wie ein normales EPROM, während die AW-Version (Associative Write) wie ein normales RAM beschreibbar ist. Anwendungsgebiet ist beispielsweise Textverarbeitung. Suchoperationen im Text (sogenanntes »Search and Replace«) werden um das bis zu 250fache beschleunigt. Eine Vizawrite-Version, die mit 27AW64 zusammenarbeitet, soll in Vorbereitung sein. (bs/hm)

Info: ProzessLogic, c/o H. Meyer, Joh.-Seb.-Bach-Str. 1, 8013 Haar, Tel. 089/4613106



DER VOLLAUTOMATISCHE HACKER

Jetzt werden auch die Hacker wegrationalisiert!

Das Programm HANS (Hackers Network Service, Bild) übernimmt nun die Hauptarbeit eines jeden Hackers — das ständige Ausprobieren von Paßwörtern und Kennungen, um sich in andere Rechnersysteme einzuschleichen.

HANS besteht aus vier Teilen: Einem Notizblock für Hacker, einem Bereich, in dem alle für die Datenfernübertragung notwendige Parameter eingestellt werden, einer Datenverwaltung für Paßwörter und als wichtigstes — eine neue Computersprache namens SHIT (Symbolic Hack-Instructions for Computer-Terms).

Damit lassen sich sogenannte Hack-Algorithmen definieren, die dem Hacker mühsame Arbeiten abnehmen, wie beispielsweise das Anwählen von Teilnehmernummern und Ausprobieren von Paßwörtern. Bis zu 30000 Paßwörter lassen sich — sortiert auf maximal hundert Dateien — dabei verwalten. HANS kann eine Datei durchprobieren oder auch alle. Zwanzig Dateien gibt es zu HANS. Laut Angabe des Herstellers läßt sich damit eine Erfolgsquote von 33 Prozent erreichen.

Das gesamte Programm ist menügesteuert und sehr anwenderfreundlich. Die zur Verfügung stehenden DOS-Operationen und das implementierte Terminalprogramm lassen nur noch wenige Wünsche offen.

Ein ausführliches Handbuch wird mitgeliefert (rund 170 Seiten). Viele Beispiele und Übungen helfen Ihnen, den Umgang mit HANS schnell zu erlernen.

Für 128 Mark erhält man ein nahezu vollständig ausgereiftes Hacker-System. Update-Versionen mit Ergänzungen zum Handbuch sollen später einmal für 10 Mark zu bekommen sein.

Übrigens: Nach neuen Gesetzesvorlagen soll das Ausprobieren von Paßwörtern strafbar sein! (B.H.P./kn)





FRAGEN ZUM C 64

(1) An der Rückseite meines C 64 befindet sich zwischen Expansion-Port und Fernsehausgang ein Regler mit der Bezeichnung »H-L« auf dem Gehäuse. Welche Bedeutung hat dieser Regler?
(2) Ich habe mich schon länger mit dem Commodore Basic V2 befaßt und möchte nun ein wenig in Maschinensprache/Assembler einsteigen. Könnten Sie mir Einsteigerliteratur dazu empfehlen?
(3) Kann man ein durch die fehlerhafte »Klammeraffen-Routine« der Floppy 1541 zerstörtes Programm wieder auf der Floppy restaurieren?
(4) Eine letzte Frage: Wer hat Erfahrungen mit dem Okimate 20-Drucker?

MARCO TRUNZER

(1) Dieser Regler ist nur bei wenigen C 64 noch vorhanden. Es handelt sich dabei um einen Kanal-Wahlschalter, mit dem der Fernsehkanal, auf dem der C 64 »sendet«, verändert werden kann. In Deutschland ist dieser Schalter, wenn überhaupt vorhanden, aus postalischen Gründen stillgelegt und abgedeckt.
(2) Wirklich ruhigen Gewissens können wir an dieser Stelle etwas Eigenwerbung betreiben: Das 64'er-Sonderheft 8/85 mit dem Titel »Assembler für Anfänger und Fortgeschrittene« ist der ideale Einstieg in die Maschinensprache-Programmierung.

Es enthält unter anderem einen 70seitigen Assemblerkurs für den Einsteiger, einen zweiten 30seitigen Kurs für Fortgeschrittene und jede Menge kleiner Utilities und Tips & Tricks zur Assemblerprogrammierung. Der besondere Clou an diesem Sonderheft: Es enthält zusätzlich einen Makro-Assembler der Spitzenklasse (Hypra-Ass) und einen sehr komfortablen Maschinensprache-Monitor (Smon) — beide zum Abtippen, aber auch auf Diskette/Kassette erhältlich.

HILFE GESUCHT

(1) Gibt es eine Möglichkeit, mittels Vizawrite erstellte Texte an einen Siemens-Fernschreiber (5-Bit-Lochstreifen) weiterzuleiten? Wie müßte eine Hardware-Lösung aussehen und wer bietet entsprechende Hardware an?
(2) Kann man ohne Anschaffung von teuren und für Sender/Empfänger ausgelegten RW/CTTY-Karten Morse- und Funkfernschreibsignale aus einem normalen Radio so umwandeln, daß sie vom C 64 verarbeitet, daß heißt, entschlüsselt und zur Anzeige auf Bildschirm oder Drucker gebracht werden können?
(3) Gibt es ein Programm, mit dem Flußdiagramme grafisch erstellt und anschließend auf Ihren logischen Aufbau hin überprüft werden?
(4) Die Software-80-Zeichen-Karten sind ohne Ausnahme unbefriedigende Lösungen. Gibt es ein Programm, das eine Darstellung von 64 Zeichen pro Zeile (also 5 Pixel pro Buchstabe) erlaubt und so eine bessere Lesbarkeit erzielt?

DIPL.-KFM. PETER GELSDORF

PROBLEME MIT EPSON UND C 128?

Ich besitze seit kurzer Zeit den neuen C 128 von Commodore und gehöre zu den sogenannten »Aufsteigern« vom C 64. Nun habe ich folgendes Problem: Beim Betrieb mit dem Drucker Epson FX-80 mit Görlitz-Interface stimmen teilweise die ausgedruckten Zeichen nicht mit den gesendeten überein.

Wer hat ähnliche Probleme und kann mit weiterhelfen? Im Fachhandel ist man zu keinerlei Auskünften bereit.

REINHARD ANDREAS SCHULZ

FRAGEN ZUM C 128

(1) Gibt es für den C 128 ein Malprogramm, bei dem man die gezeichneten Bilder über »BLOAD« laden und in eigenen Programmen verwenden kann?
(2) Kann man beim C 128 mehrere Windows definieren, oder wird bei Definition eines weiteren Windows das alte gelöscht?
(3) Wie hebt man eine Window-Definition wieder auf?
(4) Kann man in Basic (ohne POKEs) Sprites um bestimmte Winkel drehen?
(5) Kann man Sprites um beliebig viele Punkte vergrößern oder verkleinern?
(6) Wenn ein Sprite mit der PAINT-Anweisung ausgefüllt und anschließend bewegt wird, bleibt das Sprite dann ausgefüllt oder bleibt die ausgefüllte Fläche am gleichen Ort zurück?
(7) Im Markt & Technik-Buch »Das C 128 Handbuch« ist ein Listing für Grafik auf dem 80-Zeichen-Bildschirm abgedruckt. Ich weiß aber nicht die Adressen dieses Programms und kann es daher auch nicht mit »BSAVE« speichern.
(8) Wie kommt man (in Basic) an den zweiten Schreib-/Lesekopf der 1571-Floppy?

THOMAS WEBERSTAEDT

(1) Ein spezielles Malprogramm für den C 128-Modus ist uns bislang noch nicht bekannt. Sie können aber einfach ein entsprechendes Programm für den C 64 (im C 64-Modus) benutzen, um Ihre Bilder zu malen. Diese Bilder lassen sich dann völlig problemlos auch im C 128-Modus laden (die Grafik ist in beiden Betriebsarten die gleiche). Nähere Hinweise dazu erhalten Sie entweder im Handbuch zum Grafikprogramm oder auf Anfrage beim Hersteller — gegen Vorlage eines Kaufnachweises.
(2) Nein. Sobald ein Window definiert wird, ist die vorherige Einstellung vergessen.
(3) Im Direktmodus durch zweimaligem Druck auf die HOME-Taste, im Programm am einfachsten durch Definition eines neuen Windows (das natürlich auch den gesamten Bildschirm umfassen kann).
(4) Das ist nicht möglich.
(5) Nur die bekannte Verdopplung der Sprite-Größe in X- und Y-Richtung ist möglich.
(6) Die PAINT-Anweisung bezieht sich nur auf die hochauflösende Grafik. Es ist damit nicht möglich, Sprites auszufüllen.
(7) Bei dem abgedruckten Listing handelt es sich um ein Assembler-Programm. Die (vierstellige) Zahl am Anfang jeder Zeile ist die (hexadezimale) Adresse jedes Befehls. Das Programm wird mit dem eingebauten Maschinensprache-Monitor des C 128 eingegeben und gespeichert. Die Anfangs- und Endadresse des Programms ist einfach die Adresse des ersten und des letzten Befehls, also von $1400 bis $157F.
(8) Der Zugriff auf die eine oder andere Diskettenseite wird vom DOS der 1571 ganz selbsttätig geregelt. Sie brauchen keine besonderen Vorkehrungen zu treffen, um einen Schreib-/Lesekopf auszuwählen.

VIELSTIMMIGER SID

Ist der C 64 über den Audioein-/ausgang auch als Effektgerät für E-Gitarren etc. nutzbar? Wie kann man den SID auch mehr als dreistimmig benutzen? *MARKUS GURNIG*
Ausgabe 12/85

Ja, man kann den C 64 auch als Effektgerät einsetzen, da der SID einen Eingang für externe Signale hat. Hierzu muß man im SID-Register $D417 (54295) Bit 3 setzen (externe Filtereingabe ein). Danach wird das extern angelegte Signal genau wie eine Synthesizerstimme beeinflußt, so daß man mittels eines Programms verschiedene Klangfarbenverläufe einstellen beziehungsweise programmieren kann. Hierzu empfehle ich das Buch Commodore Sachbuch Band 1, Seite 323 und 452, 453.

Zur zweiten Frage: Grundsätzlich ist der SID nur dreistimmig betreibbar; allerdings könnte man ihn hardwaremäßig über den externen Signaleingang mit einem zweiten SID verbinden (siehe auch Commodore Sachbuch Band 1, Seite 453.

DIETMAR HARMS

PEEKS UND POKES MIT COMAL?

Wie kann ich feststellen, ob die vielen für den C 64 veröffentlichten Tips & Tricks auch mit der Programmiersprache Comal 80 (Steckmodul) funktionieren? Gibt es andere Möglichkeiten als das Ausprobieren? *WOLFGANG DÖRR*

Dieses Problem taucht generell bei Verwendung von Basic-Erweiterungen oder gar völlig anderen Programmiersprachen auf. In Regel weiß man ja nicht genau, welche Speicherzellen ein spezielles Programm für welche Zwecke benutzt.

Man müßte schon sehr weitgehende Unterlagen über die Speicherbelegung der Sprache haben, um Analogien zu den beim C 64-Basic möglichen Manipulationen herstellen zu können. Nur wird in den allermeisten Fällen der dazu notwendige Aufwand die vergleichsweise geringe Mühe des Ausprobierens weit übersteigen.

DRUCKER-GERÄTE-ADRESSE ÄNDERN?

Wie kann man beim 1526-Drucker einen Schalter zum Umstellen der Geräteadresse zwischen 4 und 5 realisieren? Mir fehlt ein Schaltbild des Druckers.

KARL-HEINZ HESSELBACH

AUTOSTART GESUCHT

Ich suche nach einer einfachen Möglichkeit, beim C 64 einen automatischen Programmstart zu realisieren. Die Lösung sollte keinerlei Kenntnisse über Maschinensprache voraussetzen. *DIETER RAVER*

PLOTTER-PROBLEME

Ich besitze den MC-Selbstbau-Plotter und suche ein Programm, um Platinenlayouts von »Platine 64« (Data Becker) auszudrucken. Wer kann helfen? Wer hat eine deutsche Anleitung zum Printer-Plotter 1520 und kann eine Kopie abgeben? *WOLFGANG SCHULTE*

»APFELMÄNNCHEN« MIT MPS-DRUCKER?

Wer hat ein Programm, um die Apfelmännchen-Grafik auf dem Drucker MPS 801/802/803 auszugeben? *MICHAEL NAPPE*

C 64 SCHREIBT HEBRÄISCH?

Ich beabsichtige ein Lernprogramm in hebräischer Sprache zu schreiben, die ja bekanntlich von rechts nach links gelesen wird. Wie kann ich nun den C 64/C 128 veranlassen, daß er von rechts nach links schreibt, also sämtliche Schreib- und Editiervorgänge umgekehrt zum Normalmodus durchführt? *WOLFGANG FEAER*

NETZGRAFIK MIT SEIKOSHA GP 100 VC?

Zu dem Netzgrafikprogramm aus der Ausgabe 4/85 suche ich eine passende Hardcopy-Routine für meinen Seikosha GP 100 VC-Drucker. *HELMUT SPAHN*

VON CP/M NACH MS-DOS?

Wie kann man ASCII-Dateien vom Commodore 128 auf einen IBM-kompatiblen Computer überspielen (beispielsweise eine Wordstar-Datei vom C 128 auf einen IBM-PC unter MS-DOS oder CP/M 86)?

NICO MÜLLER

Sie brauchen dazu eine serielle Schnittstelle und ein Programm zur Datenübertragung.

MPS 801-FARBBAND AUFFRISCHEN

Es hat sich ja inzwischen herumgesprochen, daß Commodore den MPS 801 zwar mit einem nachfüllbaren Tintenbehälter versehen hat, aber keine Tintenpatronen zum Nachfüllen liefert.

Ich verwende zum Nachfüllen jetzt seit sechs Monaten Stempeltusche der Firma Geha und habe seitdem überhaupt keine Probleme mehr. Zum ersten Mal seit langer Zeit kann ich mit einer Kassette tatsächlich bis zum Verschleiß des Farbbandes drucken. Geha bietet auch komplette Farbbandkassetten an (Preis etwa 16 Mark), meines Wissens nach eines der preiswertesten Angebote überhaupt mit dem zusätzlichen Vorteil gegenüber dem Original-Ersatzteil, daß die Kassetten nicht bereits beim Kauf schon ausgetrocknet sind. Ich denke, daß diese Information für viele Besitzer eines MPS 801 interessant sein könnte. *DIETER STAHLHOFEN*

Info: Geha-Farbbandkassetten und Stempeltusche zum Nachfüllen gibt es im Computer- und Büro-Fachhandel.

VIZAWRITE UND DPS 1120

Wie bekomme ich meinen DPS 1120-Drucker dazu, unter Vizawrite die deutschen Sonderzeichen auszudrucken?

KLAUS STENZEL

C 64 ALS LICHTORGEL?

Ich möchte meinen C 64 als Lichtorgel, Lauflicht etc. benutzen. Da ich aber kein Profi-Elektroniker bin, suche ich anschlußfertige Hard- und Software für diese Zwecke.

IVO GERMAN

WER KENNT SILVER REED EX42?

Wer hat Erfahrungen mit der Schreibmaschine Silver Reed EX42 und weiß, wie man sie als Drucker am C 64 verwenden kann? *BRUNO BRANDT*

CP/M FÜR COMMODORE 64

Wer hat Erfahrung mit der Übertragung von CP/M-Software anderer Computer auf das 1541-Format?

HELMUT ROESSEL

Von der Firma Bieling in Köln gibt es die nötige Hard- und Software zur Übertragung von CP/M-Programmen im Apple-Format auf das C 64-Format.

KLAUS STENZEL

Info: Heribert Bieling Hard- und Software, Grüner Hof 14, 5000 Köln 60

VIDEO-VORSPANN MIT C 64

Wer hat ein Programm zum Erstellen von Video-Filmtiteln oder kennt eine hilfreiche Adresse? *HANS RYBER, AUSGABE 2/86*

Da ich selber ein Video-Fan bin, habe ich mir über dieses Problem auch schon den Kopf zerbrochen. Meine Lösung heißt Simons Basic. Durch die Möglichkeit, Text in hochauflösender Grafik mit einfachen Befehlen selbst zu programmieren, lassen sich fast alle benötigten Effekte erzielen. Auch Laufschrift stellt mit Simons Basic kein Problem dar. *DIETER RAVER*

Die Nutzung des Computers als Titelgerät ist durchaus sinnvoll. Wir wenden dieses Verfahren schon seit einiger Zeit mit Erfolg an und sind an einem Erfahrungsaustausch mit anderen Video-Filmern oder -Gruppen zum Thema »Video und Computer« interessiert. Im Einzelfall können wir auch einen Titel in Form eines Programms »maßschneidern«, wenn uns die entsprechenden Vorstellungen und Wünsche genannt werden.

Info: Video-Workshop »Selbstgedreht«, c/o Michael Fröhlich, Kantstr. 4, 7143 Vaihingen/Enz 3

PROBLEME MIT CP/M-MODUL BEHOBEN

Mein CP/M-Modul zum C 64 funktioniert nur dann, wenn es direkt an den Computer angeschlossen ist. Beim Betrieb über eine Erweiterungsplatine läuft gar nichts mehr.

ALAIN PINEHEMAIL
Ausgabe 2/86

Ich habe einen C 64 mit CP/M-Modul und Speedbios von Oliver Joppich mit Umbau der Karte auf 56 KByte. Mit Speeddos zusammen erreiche ich eine Boot-Zeit von sechs Sekunden. Ladefehler oder ähnliche Probleme treten bei mir nicht mehr auf. Ich empfehle daher jedem Besitzer eines CP/M-Moduls, sich ebenfalls so auszurüsten.

Mit dem Bieling-Transfer-Paket gibt es keine Probleme, CP/M-Software vom Apple II auf das 1541-Format zu bekommen.

Eine weitere Erfahrung: Bei Schwierigkeiten mit dem Modul sollte man es öffnen und alle Kontaktstellen nachlöten, da die Lötstellen sehr schlecht sind. Außerdem kann bei zuviel angeschlossenen Geräten am C 64 der 6526-Baustein so belastet werden, daß er seinen Dienst verweigert (was durch Auswechseln eventuell besser wird). *WOLFGANG BÖDE*

WER KENNT DAS LIEN-YIG-LAUFWERK?

Ich bin seit einiger Zeit Besitzer eines 1541-kompatiblen Floppy-Laufwerks der Firma Lien Yig. Bei diesem Laufwerk handelt es sich um die senkrecht stehende Ausführung. Leider habe ich mit diesem Gerät viel Ärger. Solange ich normal gespeicherte Programme lade, ist alles in Ordnung. Sobald ich aber Software laden will, die mit Beschleunigern wie »Hypra-Save« gespeichert wurde, treten große Probleme auf. Dann springt das Laufwerk nicht an, läuft Amok oder bringt mich sonstwie zur Verzweiflung. Offenbar ist das DOS völlig unterschiedlich zur 1541 organisiert.

Gibt es irgendwo in Deutschland jemanden, der sich auch mit diesem nervtötenden, taiwanesischen Laufwerk abplagen muß und der mir eventuell bei meinen Problemen weiterhelfen kann? *ANDRE SCHILD*

KOPIEREN MIT 1541 UND 1571?

Kann man mit einer 1541-Floppy und dem neuen 1571-Laufwerk Disketten von einem Laufwerk zum anderen kopieren? Funktionieren alle Kopierprogramme auch mit der 1571?

DIETER STAHLHOFER

Die 1571 kann Disketten der 1541 sowohl lesen als auch selbst in diesem Format beschreiben. Das Kopieren mit zwei Laufwerken ist prinzipiell möglich. Sie müssen aber einen wichtigen Punkt beachten: Wenn Sie den C 128 in den C 64-Modus schalten, dann geht eine angeschlossene 1571-Floppy automatisch in den 1541-Modus. Leider aber ist die 1571 nur eingeschränkt kompatibel zur 1541. Die meisten Kopierprogramme (insbesondere die schnellen) laufen nicht mit der 1571. Dasselbe gilt übrigens auch für kopiergeschützte C 64-Software. Die meisten kopiergeschützten Programme laufen auf einer 1571 nicht. Das führt dann zu der sehr merkwürdigen Situation, daß Besitzer von (teuren) Originalen beim Umstieg vom C 64 zum C 128 Ihre Software nicht mehr verwenden können. Wer dagegen Raubkopien besitzt (bei denen der Kopierschutz entfernt wurde) hat keine Probleme. Raubkopien laufen auch auf der 1571 bis auf wenige Ausnahmen einwandfrei. Man sieht daran wohl recht deutlich die Kundenfreundlichkeit derartiger Kopierschutz-Mätzchen. Unsere Programm-Service-Disketten sind übrigens nicht kopiergeschützt.

64'er-DOS komplett

Schon ganz gut, aber eben nicht vollkommen könnte man zum Listing des Monats der Ausgabe 3/86 sagen, denn das Computer-ROM (Kern Gen) ist zwar vollständig, das 1541-ROM (DOS-Gen) aber leider nicht.

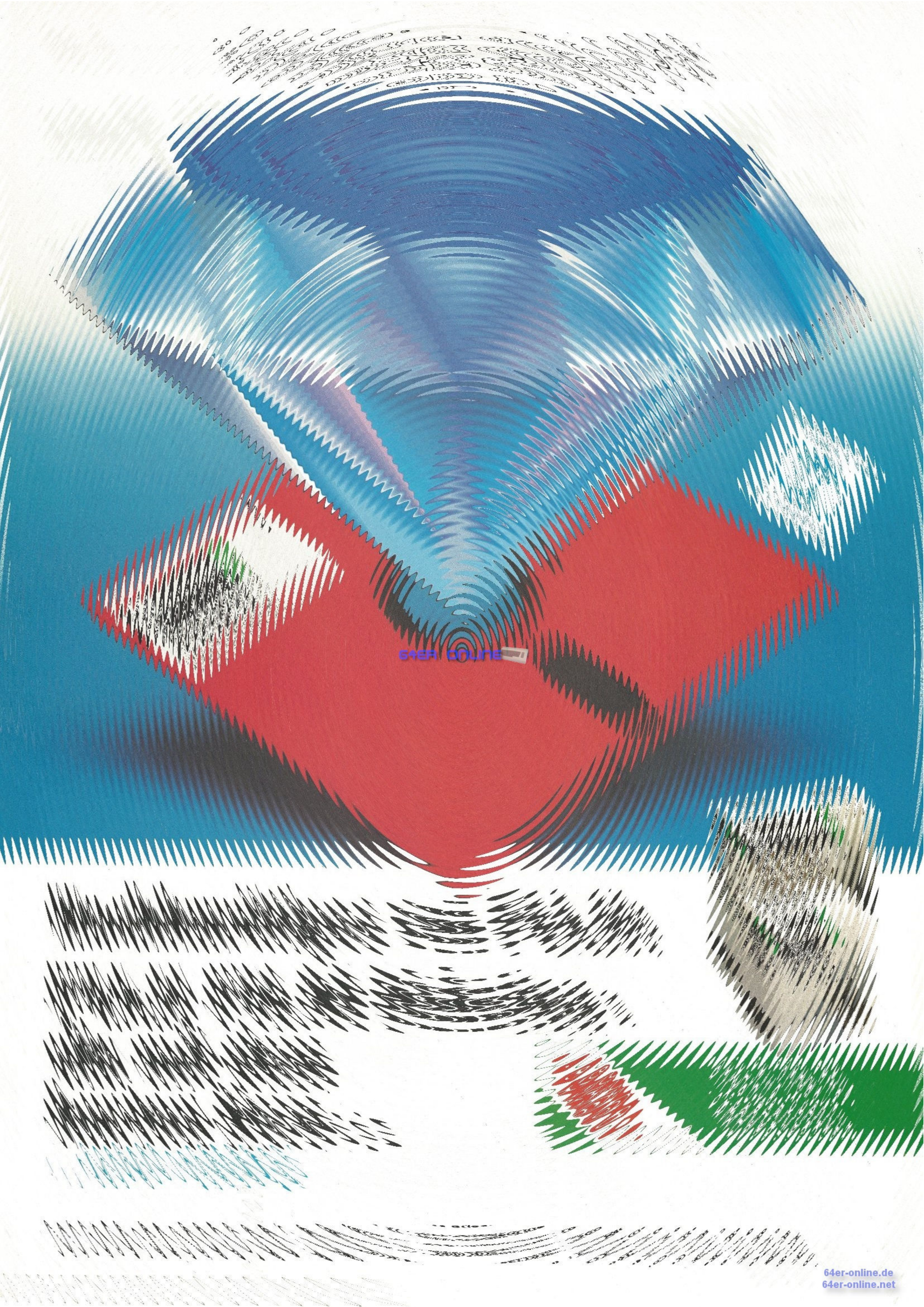
Diesmal hat er es wirklich arg getrieben — unser Fehlerteufelchen, denn er hat den bearbeitenden Redakteur nach allen Regeln der Kunst ausgetrickst. Während dieser beflissentlich darauf bedacht war, daß das Listing gut lesbar und ohne Fehler beim Zeilenumbruch abgedruckt wurde, hat das Fehlerteufelchen einfach zwei Drittel des Listings abgeschnitten.

Aber Spaß beiseite, wir entschuldigen uns dafür, daß unser Listing des Monats nicht vollständig abgedruckt wurde. Gleichzeitig holen wir das Versäumte nach und drucken das komplette Listing DOS-Gen ab. Dieses Programm erzeugt, wie in der letzten Ausgabe erläutert, das neue Floppy-Betriebssystem für das 1541-Laufwerk. In letzter Sekunde haben sich noch zwei Änderungen ergeben: Laden Sie »DOS-GEN« und geben Sie im Direktmodus Poke 2335,172 und Poke 3433,157 ein, speichern Sie »GEN-DOS«. Laden Sie »Kern Gen«, geben Sie Poke 2713,4 ein, speichern Sie Kern Gen«. Danach funktioniert alles wie beschrieben. Zum Schluß noch eine gute Nachricht: Alle diejenigen, die sich die Programm-Service-Diskette oder ein fertiges 64'er-DOS bestellt haben, brauchen sich keine Sorgen zu machen — beides wird vollständig ausgeliefert. (aw)

```
 programm : dos gen        0801 0e2f
-----------------------------------

0801 : 0b 08 c2 07 9e 32 30 36  4a
0809 : 31 00 00 00 a0 d5 a9 08  aa
0811 : 84 f7 85 f8 a0 00 a9 20  02
0819 : 84 f9 85 fa a5 ba 09 08  bf
0821 : 85 ba 20 0c ed a9 6f 20  b7
0829 : b9 ed a9 4d 20 dd ed a9  e9
0831 : 2d 20 dd ed a9 52 20 dd  0d
0839 : ed a5 f9 20 dd ed 18 a5  74
0841 : fa 69 c0 20 dd ed a9 20  58
0849 : 20 dd ed 20 fe ed a5 ba  43
0851 : 20 09 ed a9 6f 20 c7 ed  99
0859 : a0 00 20 13 ee 91 f9 c8  59
0861 : c0 20 d0 f6 20 ef ed 18  ae
0869 : 98 65 f9 85 f9 90 ad e6  8b
0871 : fa a5 fa c9 40 90 a5 a0  96
0879 : 00 b1 f7 85 f9 c8 b1 f7  9d
0881 : f0 3a 85 fa c8 b1 f7 85  54
0889 : 22 c8 b1 f7 85 23 38 98  fe
0891 : 65 f7 85 f7 90 02 e6 f8  f9
0899 : a0 00 b1 f7 91 f9 e6 f7  19
08a1 : d0 02 e6 f8 e6 f9 d0 02  d0
08a9 : e6 fa c6 22 d0 ec c6 23  d8
08b1 : 10 e8 30 c5 46 41 53 54  5e
08b9 : 20 44 4f 53 a9 08 a2 b5  0a
08c1 : a0 08 20 bd ff a2 00 a9  8d
08c9 : 20 86 c1 85 c2 a9 c1 a0  0f
08d1 : 40 4c d8 ff b7 25 0c 00  42
08d9 : b6 34 27 45 52 20 44 4f  f2
08e1 : 53 20 56 31 83 26 02 00  72
08e9 : ea ea 8b 26 02 00 ea ea  92
08f1 : 80 27 23 00 60 29 03 85  34
08f9 : 4a a9 d2 a4 22 f0 08 98  5c
0901 : 49 ff 4a e9 00 18 0a 0a  17
0909 : e5 4a 60 85 79 2c 85 7a  e0
0911 : 8d 0a 1c 4c 02 e9 60 8d  40
0919 : 28 04 00 4c 97 e7 4c 97  e6
0921 : 28 04 00 4c 9a e7 ea 16  96
0929 : 29 09 00 29 08 85 98 ad  67
0931 : 0a 1c f0 14 20 29 17 00  b0
0939 : 87 ff a9 04 2c 00 18 f0  b0
0941 : fb 0a 0d 00 18 20 9f e9  5a
0949 : 4c 87 e9 20 b7 e9 4a 3a  40
0951 : 29 0a 00 29 01 d0 f9 90  44
0959 : 07 a5 98 d0 09 45 29 20  13
0961 : 00 59 ea 20 c0 e9 4a 90  72
0969 : f7 20 ae e9 20 59 ea 20  12
0971 : c0 e9 29 01 d0 f3 a9 08  f4
0979 : 85 98 20 c0 e9 29 01 d0  f8
0981 : 34 7c 29 01 00 00 86 29  ca
0989 : 01 00 d7 c9 29 11 00 4c  6d
0991 : 20 ff 20 59 ea 29 04 d0  8e
0999 : f9 20 9c e9 29 01 d0 f9  d8
09a1 : 29 2a 02 00 c9 ff a3 2a  df
09a9 : 04 00 00 4c 10 ff 87 2b  ac
09b1 : 01 00 a2 89 2b 01 00 86  54
09b9 : 8b 2b 08 00 8e 3b 02 e8  79
09c1 : 86 ac a9 ff 94 2b 10 00  ea
09c9 : ad 85 b4 a9 84 8d 3a 02  3d
09d1 : a2 0f 8e 56 02 e8 86 1e  27
09d9 : a5 2b 01 00 88 a7 2b 01  c9
09e1 : 00 f7 a9 2b 0f 00 e0 8d  3c
09e9 : 4f 02 a9 ff 8d 50 02 a9  5a
09f1 : 01 85 1c 85 1d b9 2b 2e  15
09f9 : 00 f6 20 63 cb 20 fa ce  30
0a01 : 20 59 f2 a9 22 85 65 a9  f7
0a09 : eb 85 66 a9 0a 85 69 a9  4b
0a11 : 05 85 6a a9 73 20 c1 e6  b6
0a19 : a9 00 8d 00 18 a9 1a 8d  78
0a21 : 02 18 2c 00 18 30 1d 3f  30
0a29 : 2c 0e 00 a5 7f 85 86 c8  e1
0a31 : 84 7f a5 1c f0 15 20 13  c0
0a39 : d3 4e 2c 24 00 1c 10 0e  00
0a41 : a9 12 85 10 a9 01 85 11  32
0a49 : 85 1c a9 b0 85 05 a9 01  86
0a51 : 85 1d a5 86 85 7f ad 00  aa
0a59 : 1c 29 f7 a6 6f f0 02 09  75
0a61 : 08 c2 2c 01 00 08 ce 2c  c9
0a69 : 03 00 2d 78 00 5f 32 0b  a1
0a71 : 00 d2 45 65 8d 00 1c ad  7d
0a79 : 0c 1c 29 fe 6b 32 01 00  0a
0a81 : ee 7a 32 01 00 25 a5 32  7d
0a89 : 01 00 03 a9 32 01 00 01  ae
0a91 : ad 32 01 00 10 84 33 06  96
0a99 : 00 4f 54 44 20 [illegible] e7 ea  62
0aa1 : 37 02 00 4a 4a ed 37 19  46
0aa9 : 00 85 52 b1 30 29 07 0a  b3
0ab1 : aa c8 d0 06 a5 4e 85 31  f9
0ab9 : a4 4f b1 30 0a 90 01 e8  72
0ac1 : 0a 8a 07 38 22 00 aa bd  22
0ac9 : c0 f8 a6 52 1d a0 f8 85  bf
0ad1 : 52 b1 30 29 3f 4a 85 53  30
0ad9 : c8 b1 30 6a 4a 4a 4a aa  49
0ae1 : bd c0 f8 a6 53 1d a0 f8  a4
0ae9 : 2a 38 01 00 53 30 38 2e  64
0af1 : 00 aa c8 b1 30 10 01 e8  08
0af9 : 29 7c 4a 4a 85 54 bd a0  6f
0b01 : f8 a6 54 1d c0 f8 85 54  98
0b09 : b1 30 29 03 85 55 c8 d0  45
0b11 : 06 a5 4e 85 31 a4 4f b1  07
0b19 : 30 0a 29 c0 05 55 2a 60  15
0b21 : 38 40 00 aa b1 30 29 1f  4e
0b29 : c8 84 34 a8 bd a0 f8 19  4c
0b31 : c0 f8 85 55 a4 34 60 fa  dd
0b39 : fa fa fa fa a5 43 c5 4c  f2
0b41 : d0 03 4c 36 fc 20 0e fe  74
0b49 : a2 00 8e 24 06 a2 05 8e  ba
0b51 : 25 06 a9 ff 8d 01 1c 50  d6
0b59 : fe b8 ca d0 fa 20 00 fe  2f
0b61 : 4c 98 fb 8b 39 01 00 16  31
0b69 : ae 39 01 00 81 48 3a 01  39
0b71 : 00 02 bb 3a 01 00 02 ca  56
0b79 : 3a 0b 00 14 a9 60 85 20  af
0b81 : ad 00 1c 20 81 e7 d6 3a  61
0b89 : 01 00 4a d8 3a 01 00 01  e6
0b91 : da 3a 23 00 22 85 51 d0  86
0b99 : 0a a0 00 84 ff d1 32 f0  bd
0ba1 : 0a 91 32 a9 0a 8d 20 06  cf
0ba9 : 4c 9c f9 ad 00 1c 29 10  1d
0bb1 : d0 05 a9 08 4c db fd fe  09
0bb9 : 3a 1d 00 79 f8 00 80 4f  e1
0bc1 : 44 40 c0 ca 86 98 a9 08  93
0bc9 : 8d 00 18 20 8a ff a9 00  b0
0bd1 : 20 e5 cd 20 3e de be 99  64
0bd9 : 1c 3b 8b 00 ca 86 10 e6  65
0be1 : 1c a5 81 2c a9 4f 85 11  03
0be9 : a5 80 f0 4b c9 24 b0 0d  0e
0bf1 : 85 0a a6 81 20 4b f2 c5  09
0bf9 : 81 f0 02 b0 06 20 75 fb  b8
0c01 : 4c 4d d5 a9 e0 85 02 a9  34
0c09 : e2 85 21 a2 02 8d 5d 02  51
0c11 : 8d 98 02 20 99 d5 c9 02  e2
0c19 : 90 ca a9 c0 c5 21 f0 0f  d8
0c21 : 85 02 85 21 a5 02 30 fc  52
0c29 : ad 5d 02 85 02 d0 dc 20  10
0c31 : 75 fb 4c 3f d6 a9 ff 4c  f2
0c39 : 8a ff a9 d0 8d 05 18 2c  01
0c41 : 05 18 10 26 2c 00 1c 30  af
0c49 : f6 ad 01 1c b8 a0 00 60  2b
0c51 : 00 80 4f 44 40 c0 ea ea  79
0c59 : ad 00 1c 10 12 50 f9 b8  0c
0c61 : ee 25 06 d0 f3 ee 24 a8  16
0c69 : 3b 0b 00 10 ee a9 03 4c  0d
0c71 : db fd a5 43 85 4c b4 3b  21
0c79 : 01 00 e0 e1 3b 01 00 63  71
0c81 : aa 3c 01 00 a2 ac 3c 01  0c
0c89 : 00 86 af 3c 01 00 10 b1  f3
0c91 : 3c 07 00 a2 06 a9 ff 8d  6e
0c99 : 01 1c c2 3c 09 00 b9 00  58
0ca1 : 03 8d 01 1c 50 fe b8 d1  b2
0ca9 : 3c 08 00 a9 55 8d 01 1c  1d
0cb1 : 50 fe b8 db 3c 0b 00 fa  3c
0cb9 : a2 05 a9 ff 8d 01 1c 50  3a
0cc1 : fe b8 e8 3c 02 00 fa a0  2a
0cc9 : eb 3c 0a 00 b9 00 01 8d  10
0cd1 : 01 1c 50 fe b8 c8 f7 3c  fe
0cd9 : 2d 00 b1 30 8d 01 1c 50  6b
0ce1 : fe b8 c8 d0 f5 a9 55 8d  a5
0ce9 : 01 1c ce 28 06 f0 13 ae  43
0cf1 : 26 06 50 fe b8 ca d0 fa  29
0cf9 : a5 32 18 69 0a 85 32 4c  19
0d01 : b1 fc 50 fe b8 50 fe 28  7f
0d09 : 3d 01 00 01 3a 3d 02 00  7d
0d11 : 7a fb 63 3d 02 00 7a fb  0b
0d19 : b6 3d 01 00 20 0e 3e 16  46
0d21 : 00 a2 1f ad 0c 1c 29 1f  74
0d29 : 09 c0 8d 0c 1c a9 ff 8d  a1
0d31 : 03 1c a9 55 8d 01 1c 2d  03
0d39 : 3e 01 00 10 8b 3e 01 00  a9
0d41 : 58 10 3f 8a 00 8e 03 18  73
0d49 : a9 02 8d 00 18 a9 1a 8d  a9
0d51 : 02 18 4c a7 ea 2c 00 18  a8
0d59 : 30 5b ad 0a 1c f0 56 a9  d9
0d61 : 7a 8d 02 18 ad 00 18 29  b3
0d69 : fd 8d 00 18 a9 04 2c 00  9b
0d71 : 18 d0 fb ea ea ea ae 00  0e
0d79 : 18 bd 00 fb 4a 4a ea ae  ef
0d81 : 00 18 1d 00 fb 4a 4a ae  6d
0d89 : 00 18 1d 00 fb 4a 4a ea  ee
0d91 : ae 00 18 1d 00 fb 85 85  ea
0d99 : a9 02 0d 00 18 8d 00 18  a5
0da1 : 29 04 d0 04 85 f8 a9 04  50
0da9 : 2c 00 18 30 6d a9 1a 8d  89
0db1 : 02 18 a5 85 60 a9 08 85  58
0db9 : 98 4c cc e9 bd 3e 02 85  c8
0dc1 : 85 ad 00 18 29 f7 8d 00  a8
0dc9 : 18 a5 85 4a 4a 4a 4a 9b  b6
0dd1 : 3f 4d 00 a9 01 2c 00 18  8e
0dd9 : 30 41 d0 f9 8e 00 18 8a  7b
0de1 : 0a 29 0f 8d 00 18 ad 85  78
0de9 : 00 29 0f 8d 00 18 0a 29  2f
0df1 : 0f ea 8d 00 18 a9 08 a6  15
0df9 : 98 d0 02 09 02 8d 00 18  58
0e01 : 60 2c 00 18 10 12 a5 23  e9
0e09 : 05 f8 d0 0c a2 ff 8e 0a  b8
0e11 : 1c a9 80 ca d0 fb 85 f8  70
0e19 : 4c a5 e9 85 05 4c 5b e8  55
0e21 : dc 29 03 00 8d 05 18 e6  82
0e29 : 3e 01 00 0e 00 00 02 99  e5
```

Listing. DOS-Gen, jetzt komplett.


64ER ONLINE

64'er Test

Microline 182 – der schafft Platz

Der Microline 182 ist direkt an den C 64 anschließbar. Trotzdem muß man nicht auf die Vorteile eines grafik- und umlautfähigen Druckers verzichten.

Der Microline 182 (Bild 1) ist ein 9 x 9 Matrixdrukker, der mit einer Centronics-Schnittstelle oder mit dem seriellen IEC-Bus des C 64 geliefert wird.

Nachdem das Interesse am Microline 182 geweckt war, galt es, die Farbbandkassette einzulegen. Ohne jede Mühe und mit sauberen Fingern läßt sich die Aufgabe erledigen. Die Kassette wird über den Druckkopf des Microline 182 geschoben und mit einem leichten Druck eingerastet. Dabei sitzen Kopf und Kassette auf einem beweglichen Schlitten, der sauber auf zwei Schienen gelagert ist. Bei genauerem Hinsehen stellt man fest, daß sich noch ein kleiner Motor auf diesem Schlitten befindet. Über ein Getriebe greift ein Zahnrad in eine horizontal am Boden des Druckers angebrachte Zahnstange.

Die zweite Überraschung gab es beim Einlegen des Papiers: Einschieben, am Handrad drehen — fertig. Ohne Verheddern und schrägen Einzug. Die Walzenanordnung ist ähnlich einer Schreibmaschine, wobei die Stachelwalzen direkt in die Schreibwalze integriert sind. Ebenso leicht lassen sich Einzelblätter, Rollen- oder Endlospapier einlegen. Die Stachelwalzen lassen sich einfach und sicher den Gegebenheiten anpassen und sind nach dem Feststellen unverrückbar arretiert. Für alle, die ihren Drucker auf einen Ständer stellen möchten, haben die Konstrukteure auf der Geräteunterseite einen ausreichend großen Schlitz angebracht, um das Papier von unten zuzuführen.

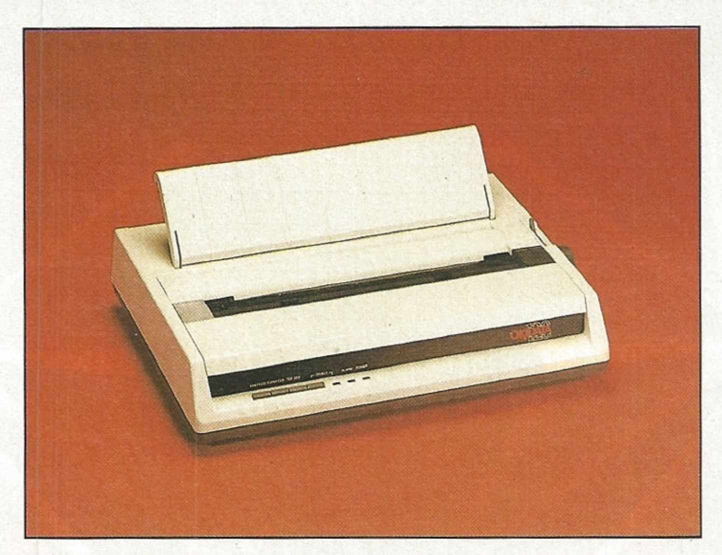


Bild 1. Ein kompakter Drucker — der Microline 182

Mit acht DIP-Schaltern werden die Papierlänge eingestellt, einer aus neun Zeichensätzen ausgewählt, die Geräteadresse eingestellt und zwischen normaler oder einer höheren Auflösung im Grafikmodus entschieden. Zugänglich ist die Schalterreihe nach Entfernen einer Schraube und Abnehmen eines kleinen Deckels an der Oberseite.

Die von dem Referenzdrucker der Preisklasse II Star (siehe Artikel »So testen wir« in dieser Ausgabe) gewohnte Vielfalt an Schriftarten und Darstellungsformen findet man beim Microline 182 nicht. Allerdings reicht das Angebot für den üblichen Druckeralltag aus. Pica, Elite, Schmal- und Breitschrift, Fettdruck, Hoch- und Tiefstellen sowie Negativschrift sind als Features vorhanden (Bild 2 und 3). Auch kann mittels ESC-Befehlen zwischen den Zeichensätzen hin- und hergeschaltet werden. Gerade diese Zeichensätze sind es, die den Microline 182 von anderen Druckern unterscheiden. Während man bei vielen Druckern, die sich anstrengen, möglichst gut an den C 64 angepaßt zu sein, keine Umlaute findet, hat der ML 182 sie. In einem der zehn Zeichensätze wurde einfach auf ein paar (unwichtige) Grafikzeichen verzichtet und der Zeichensatz an den richtigen Stellen mit den deutschen Sonderzeichen ausgestattet.

Normal PICA
Normal ELITE
Schmale Schrift
Fettdruck 123
Reverse Schrift
Unterstrichen
Hochgestellt
Chemisch H_2O
$y_1 = x^2+4x-12$
BREIT

Bild 2. Das Schriftbild reicht für viele Anwendungen aus

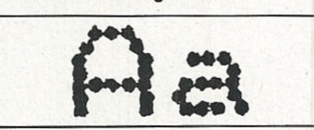


Bild 3. Fünffache Vergrößerung

Der Microline 182 ist im Gegensatz zum MPS 802 grafikfähig. Im »Bit Image Modus« kann er entweder 60 oder 120 Punkte pro Inch darstellen (Tabelle). Damit ist er zwar kein ausgesprochener Zeichenkünstler, aber für einfache Hardcopys zum Beispiel mit Print Shop, Simons-Basic oder Hi-Eddi reicht die Grafikfähigkeit vollkommen aus. Ein wesentlicher Vorteil der Centronics-Version des ML 182, nämlich seine NLQ-Schrift, wurde bei der Commodore-Version weggelassen.

Qual der Wahl

Wer sich mit dem Gedanken trägt, seinen C 64 um einen ML 182 zu erweitern, hat keine leichte Wahl. Er kann sich zwischen einer Version mit Centronics-Interface und NLQ-Schrift aber ohne Commodore-Anpassung und einer Version mit bestens gelungenem Commodore-Befehlssatz, aber ohne NLQ-Schrift entscheiden. Auch der Preis hilft dabei nicht weiter, denn beide Modelle kosten jeweils 1099 Mark. Für welche Variante man sich letztendlich entscheidet, kann nur durch ein genaues Abschätzen der eigenen Wünsche entschieden werden. (E. Konther/aw)

Info: Okidata GmbH, Emanuel-Leutze-Str. 8, 4000 Düsseldorf 1

Tabelle. Der Microline 182 in der Übersicht

Name des Druckers :	Microline 182	empfohlener Preis :	1099 Mark incl.
Unterstreichen :	Ja	Proportionalschrift :	nein
Zeichenmatrix :	9 x 9	NLQ-Matrix :	—
Papierbreiten :	127 — 254 mm	Zeichenvorrat :	2 CBM + 8 intern.
Papierarten :	Endlos, Rolle, Einzelblatt	Durchschläge :	3
Zeichen/Zeile :	bis 137	Selbsttest :	ja
Hexdump :	nein	Autom. Einzelbl.:	nein
Pufferspeicher :	nein	Rückwärtstr. :	nein
Ladb. Zeichensatz PROOF :	nein	Probetext :	2 : 46 Minuten
Geschwindigkeit :	120 ang. 110 gem.	NLQ-Geschwin. :	—
Grafikmodi :	60 und 180 Punkte pro Inch		
Funktionstasten :	LF, FF, TOF, SELECT		
Ausstattung :	Handbuch (engl.) Papierseparator, Farbband		
Schriftarten :	Pica, Elite, Revers, Hoch, Tief, Breit, Fett		
Sonderfunktionen :	—		

Nach dem großen Erfolg des SG-10 setzt Star seine Reihe der hochwertigen Drucker mit dem neuen NL-10 fort. Wir haben ihn getestet und finden, daß er ausgezeichnet zu Commodore-Computern paßt — lesen Sie warum.



Bild 1. Der Star NL-10, ein echter Star

64'er Test

Star NL-10 — ein Drucker für Sie

Das Jahr 1986 verspricht eines der interessantesten in der Druckergeschichte zu werden. Etwas mehr als ein Jahr nach der Vorstellung des Star SG-10, der damals in Fachkreisen für Furore sorgte, annonciert Star das Nachfolgemodell, den NL-10 (Bild 1). Obwohl der offizielle Vorstellungstermin auf die CeBIT 86 datiert ist, stand der 64'er-Redaktion bereits Ende Januar eines der ersten Geräte in Deutschland zum Test zur Verfügung.

Jeder, der sich heute für einen neuen Drucker interessiert, gehört zu den glücklichen, die in den Genuß einer nie dagewesenen Preis-Leistungs-Schere kommen. Gemeint ist damit das ständige Auseinanderdriften von Preis und Leistung. Was für die Druckerhersteller das Signal zum Anlegen härterer Bandagen im Kampf um Marktanteile ist, bedeutet für den Anwender, daß er immer mehr Druck-Power für immer weniger Geld erhält. Was noch vor wenigen Jahren auf die Computer beschränkt schien, (die Druckerpreise blieben lange konstant) hat nun auch den Druckermarkt erfaßt. Preise, die anbetrachts des hohen mechanischen Anteils in jedem Drucker als unmöglich galten, sind mittlerweile Realität.

Auch an Star ist diese Entwicklung nicht unbeachtet vorübergegangen. Obwohl der NL-10 ein wesentlich erweitertes Leistungsspektrum gegenüber seinem Vorgänger hat, liegt sein Listenpreis mit 1145 Mark einiges unter dem des SG-10. Aber was für Leistungen sind das? Was darf man erwarten? Besser sollte man die Frage danach stellen, was man sich wünscht. Viele Leserbriefe zeugen davon, daß der ideale Drucker möglichst direkt anschließbar, aber trotzdem auch an anderen Computern betreibbar sein soll, daß er problemlos zu bedienen, möglichst viele Schriften beherrschen und vor allem grafikfähig sein soll. Der Wunsch nach der NLQ-Schrift (Near Letter Quality gleich Schönschrift) wird gleichzeitig immer häufiger genannt. Letzter, aber sicherlich nicht unwichtigster Wunsch, ist das möglichst reibungslose Zusammenspiel des Druckers mit verschiedenen Daten-, Grafik- und Textprogrammen. Verständliche Wünsche, wenn man bedenkt, daß ein Drucker eigentlich das machen sollte was der Mensch will und nicht umgekehrt.

Der große Wurf

Mit dem NL-10 hat Star versucht, alle diese Wünsche zu erfüllen. Es hat allen Anschein, als ob es ihnen auch gelungen ist. Betrachten wir den NL-10 etwas genauer. Das Gehäuse ist beigefarben und paßt exzellent zum C 128, aber auch neben dem C 64 gefällt die Farbe. Wichtiger als die Farbe aber ist die Art, wie der NL-10 Papier verarbeitet. Während es beim SG-10 noch ein kleines Ärgernis war, daß durch den oben angebrachten Traktor jedesmal beim Einspannen ein Blatt verlorenging, hat der NL-10 den Traktor hinter der Schreibwalze im Gehäuse versenkt. Die Stachelwalzen lassen sich auf der ganzen Breite des Druckers verschieben, so daß auch schmale Etiketten problemlos eingespannt werden können. Links neben der Schreibwalze, die unter einer flachen Plastikabdeckung, die beim Anheben übrigens den Druckvorgang anhält, verborgen ist, befindet sich ein Multifunktionshebel. Seine Aufgabe ist es, einzustellen, ob Einzel- oder Endlospapier eingespannt ist. Seine zweite Funktion ist, den automatischen Einzelblatteinzug zu aktivieren. Ganz gleich mit welcher Papierart gedruckt werden soll, man legt einfach den Hebel nach hinten und das Papier wird automatisch eingezogen, zentriert und hinter den Andruckrollen fixiert. Der 9-Nadel-Druckkopf macht ebenso wie seine Führungsschienen einen sehr soliden und dauerhaften Eindruck. Links und rechts ne-

Star NL-10
NLQ-Schönschrift
Normalschrift
Eliteschrift
Schmalschrift
Proportional
Breit
Fettschrift
Hervorgehoben
Reverseschrift

Bild 4. Die große Auswahl — Schriftarten



Bild 2. Die NLQ-Schrift in fünffacher Vergrößerung

Tabelle. Das Profil des NL-10

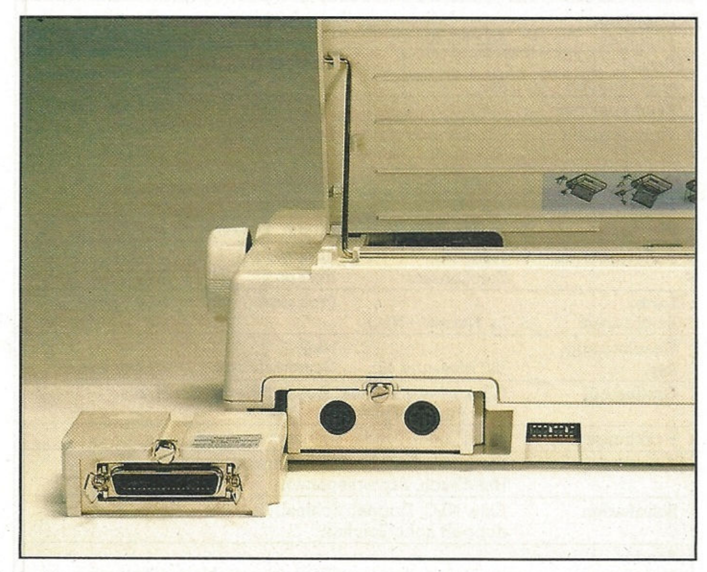
Name des Druckers	Star NL-10	empfohlener Preis	1145 Mark inkl.
Unterstreichungen	Ja	Proportionalschrift	Ja
Zeichenmatrix	9 x 11	NLQ-Matrix	18 x 23
Papierbreiten	60—270 mm	Zeichenvorrat	CBM, ASCII, CBM + Umlaute
Papierarten	Einzel-Endlos	Durchschläge	2 + Original
Zeichen pro Zeile	bis 137	Selbsttest	Ja
Hexdump	Ja Hex + ASCII	Autom. Einzelbl.	Ja
Pufferspeicher	Nein, nur Zeichensatz	Rückwärtstransport	Ja
Ladb. Zeichensatz	Ja, Normal + NLQ	Probetext	2:50 Minuten
Geschwindigkeit	120 Messung 121	NLQ-Geschwindigk.	30 Zeichen/s
Grafikmodi	CBM 7-Nadel-Grafik, Epson 8-Nadel-Grafik bis 1920 Punkte/Zeile		
Funktionstasten	LF, FF, Bold, Mode, Online		
Ausstattung	Modul nach Wahl (CBM, Centronics, IBM), Handbuch, Papierseparator		
Schriftarten	Elite, NLQ, Doppel, Schmal, Fett, Breit, Hoch, Tief, doppelt hohe Zeichen		
Sonderfunktionen	Randeinstellung per Tastendruck, Modus-Fixierung		

ben dem Druckkopf findet man etwas, das man normalerweise nur bei einer Schreibmaschine erwartet. Auf zwei Formstücken aus klarem Plastik sind hilfreiche Justierstriche angebracht, die besonders beim Ausfüllen von Formularen von Nutzen sind. Wer beim NL-10 eine Abrißkante für das Papier sucht, braucht dies nicht vergeblich zu tun, denn an der Abdeckung befindet sich eine Kante, die ihren Namen zu recht trägt, denn sie ist messerscharf.

Der NL-10 arbeitet mit einer breiten Nylon-Farbbandkassette, die sich ohne Farbabdrücke auf den Fingern zu hinterlassen, einlegen läßt. Auch die DIL-Schalter gaben keinen Grund zur Kritik, denn sie befinden sich gut erreichbar auf der Gehäuserückseite.

Die NL-10-Lightshow

Der NL-10 läßt sich wie ein Drucker der Spitzenklasse programmieren. Gut erreichbar auf der vorderen Gehäuseoberseite besitzt er fünf verschiedene Schalter und sieben Leuchtdioden. Mit den Schaltern lassen sich neben den üblichen Funktionen wie On Line, Zeilen- und Seitenvorschub die gewünschten Schriftarten bis hin zur exzellenten NLQ-Schrift (Bild 2) einstellen. Die fünfte Taste dient der Wahl der Fettschrift in Kombination mit der gewünschten Schriftgröße. Im Off Line-Modus kann man per Tastendruck sogar die beiden Ränder einstellen. Der Druckkopf wird dabei von links oder rechts so lange in Mikroschritten bewegt, bis die gewünschte Position erreicht ist. Eine genaue Justierung auf besondere Formate und Formulare wird damit zum Spiel statt zum streßbeladenen Manöver. Der Clou der Tastenprogrammierung ist die Möglichkeit, die gewählte Einstellung zu fixieren. Natürlich lassen sich sämtliche Funktionen auch über die bekannten CHR$- und ESC-Befehle einstellen. Dabei wartet der NL-10 mit einer nützlichen und einer netten Raffinesse auf. Zunächst die nette. Hat man den Drucker fleißig mit verschiedenen Schriften programmiert, so flackern bei jedem Schriftwechsel die zugehörigen Leuchtdioden kurz auf. Wesentlich wichtiger aber ist der Befehlssatz des NL-10. Dieser Befehlssatz wird aber im wesentlichen durch das verwendete Schnittstellenmodul beeinflußt. Beim Kauf hat man die Auswahl zwischen einem Centronics- (ESC/P), einem IBM- und erfreulicherweise auch einem Commodore-Modul für C 64/C 128 (Bild 3). Im Preis von 1145 Mark ist ein Modul nach Wahl einbegriffen. Mit dem Commodore-Modul, das einfach auf der Geräterückseite eingesteckt wird, ist der NL-10 direkt an den C 64, beziehungsweise C 128 anschließbar. Trotzdem bleibt die Möglichkeit, bei Bedarf ein Centronics- oder IBM-Modul (je 150 Mark) kaufen zu können. Der NLQ-10 ist wohl der einzige Drucker, bei dem die Programmierung des Commodore-Moduls so gut gelungen ist, daß man als C 64-, C 128-Besitzer getrost den Drucker mit entsprechendem Commodore-Modul kaufen kann. Trotz der Implementierung aller Befehle des MPS 803, einschließlich der Grafikbefehle, bleiben die Funktionen beispielsweise eines Epson LX-80 erhalten. Gleiches gilt für den Zeichensatz, hier hat man die freie Auswahl. Zum einen kann man den original Commodore-Zeichensatz oder den ASCII-Zeichensatz verwenden. Einer der DIL-Schalter ermöglicht es sogar, den Commodore-Zeichensatz an Stelle einiger Grafikzeichen um die deutschen Sonderzeichen und Umlaute zu bereichern. Das ganze haben wir mit verschiedenen Text-und Grafikprogrammen getestet. So druckt der NL-10 beispielsweise problemlos mit Vizawrite 64 und Textomat plus sowie ProText zusammen. Jedes erstellte Schriftstück hat alle Umlaute plus fast alle Commodore-Grafikzeichen. Dabei war es in keinem Fall notwendig, auch nur einen einzigen Befehl vorher an den Drucker zu übermitteln — alles funktioniert auf Anhieb! Bei den Grafikprogrammen war das Ergebnis sogar noch besser. Sowohl mit dem Copy-Befehl von Simons Basic (HiRes-Hardcopy), als auch mit Print Shop läßt sich ohne Anpassungsprobleme sofort drucken. Wohlgemerkt, das alles funktioniert, ohne auch nur einen Befehl aus dem übrigens sehr guten deutschen Handbuch wissen zu müssen. Bei diesen Programmen verwendet der NL-10 die 7-Nadel-Grafik des MPS 803. So richtig zum Zug kommt er aber erst, wenn man seine LX-80-Fähigkeiten verwendet. Dann ist er in der Lage, Grafiken mit bis zu vierfacher Dichte (wie der Fujitsu DX 2100) zu drucken. Mit dieser Grafik funktionieren übrigens alle Hardcopy-Routinen für Epson-Drucker. Nicht unerwähnt bleiben soll auch der Hexdump-Modus, bei dem der NL-10 sowohl Hex-Zahlen als auch in Klarschrift deren Dezimalwerte darstellt.


Bild 3. Der NL-10 — flexibel durch Module

Auch für Programmierer

Wer lieber seine eigenen Buchstaben entwerfen will, hat dazu ausreichend Gelegenheit, denn sowohl die Standard-, als auch die NLQ-Zeichen sind vor keiner Veränderung sicher. Seine exzellenten Schriftqualitäten (Bild 4) prädestinieren den NL-10 zur Textausgabe. Trotzdem besitzt er auch Fähigkeiten, die das Herz jedes Programmierers höher schlagen lassen. Der NL-10 druckt jedes Commodore-Zeichen korrekt, so wie es auch auf dem Bildschirm zu sehen ist. Mit den Cursor- und Farbsteuerzeichen macht er aber etwas ganz besonderes. Ein Listing, mit dem NL-10 erzeugt, sieht automatisch ähnlich aus, wie unsere Checksummer-Listings, denn alle wenig aussagekräftigen Steuerzeichen werden in Klarschrift übersetzt. Die manchmal schon nervtötende Suche in verschiedenen Vergleichstabellen findet mit dem NL-10 endgültig ihr Ende.

Wachablösung

Was macht man mit einem Drucker, der eigentlich nur wenige Wünsche offen läßt? Klar — man macht ihn zum Standard. Sicherlich wäre es wünschenswert, wenn der NL-10 etwas mehr als 120 Zeichen pro Sekunde (NLQ 30 Zeichen/Sekunde) schaffen würde. Auch eine Farboption würde ihn nochmals aufwerten. Im Vergleich zur Konkurrenz in dieser Preisklasse steht der NL-10 allerdings einsam an der Spitze. Deshalb löst der NL-10 auch den SG-10 als Referenzdrucker der Preisklasse II ab (siehe auch Artikel »So testen wir« in dieser Ausgabe).

Es sind seine rundum positiven Fähigkeiten (Tabelle), die eine Flexibilität in jede Richtung garantieren und der problemlose Einsatz mit allen professionellen Programmen, die den NL-10 zum idealen Drucker für Commodore-Computer machen. (aw)

Info: Star Micronics, Deutschland GmbH, Frankfurter Allee 1-3, 6236 Eschborn/Ts., Telefon: 06196/70180

So testen wir – das sind unsere Referenzdrucker

Wer Messungen durchführt, muß einen Maßstab haben. Das gilt fürs Landvermessen ebenso wie für einen Hardware-Test. Doch wie findet man einen geeigneten Maßstab bei einer so vielfältigen Materie? Ganz einfach – man nimmt das beste was man bekommen kann – unsere Referenzdrucker waren geboren.

Es sind nun genau zwei Jahre, seit es Ihre 64'er gibt, sie wird von Fachredakteuren gemacht, die sich seit vielen Jahren mit der Materie Computer und allem was dazugehört beschäftigen. Im Laufe der Zeit sammelt so ein Redakteur (Bild 1) eine Menge Erfahrung. Daraus entwickeln sich Meß- und Testverfahren, die zum einen einen objektiven Vergleich verschiedener Produkte und zum anderen ein möglichst präzises Testergebnis ermöglichen. So haben wir beispielsweise einen Drucker-Test entwikkelt, der versucht, belegbare Meßdaten (Labortest) mit dem tatsächlichen Nutzen des Gemessenen (Praxistest), zu vereinigen. Dieses Testverfahren wollen wir Ihnen kurz beschreiben. Jeder Drucker, der getestet werden soll, muß zunächst über unsere Meßstrecke. Dabei wird festgestellt, wie schnell er tatsächlich ist ohne Beeinflussung durch Programm, Schnittstelle oder Zeilenvorschub im reinen Zeichendruck. Falls der Drucker einen NLQ-Modus besitzt, wiederholen wir diesen Test mit der NLQ-Schrift (dann allerdings durch den Computer gesteuert). Diesen Wert vergleichen wir mit den Angaben des Herstellers. Falls sich eine Abweichung ergibt, finden Sie diesen Wert in der Tabelle und/oder dem Text wieder. Der nächste Test besteht aus einem genau 8 KByte langen Probetext, der alle die Geschwindigkeit beeinflussende Faktoren berücksichtigt. Dazu gehören das Drucken von einzelnen Zeichen (je eines am linken und rechten Rand), von halben Zeilen (Drucklogiktest), ganzen Zeilen mit verschiedenen und Zeilen mit gleichen Buchstaben (wichtig für Typenraddrucker, um die Drehgeschwindigkeit zu ermitteln) und einem Absatz von Fließtext mit Sonderfunktionen. Der letzte Teil des Tests besteht aus hundert Zeilenvorschüben, denn auch die sind wichtig. Die so ermittelte Druckzeit stellt einen Durchschnitt über alle Leistungen des Druckers dar. Wir halten es für wenig sinnvoll, jedes Einzelergebnis des Probetextes zu veröffentlichen, denn wer druckt normalerweise schon 1000mal den Buchstaben »A« in einer Zeile. Es ist vielmehr wichtig, eine Vergleichszahl zu finden, die es ermöglicht, verschiedene Drucker auch über längere Zeiträume miteinander zu vergleichen. Der letzte Teil unseres Tests besteht darin, daß der Redakteur seinen Redaktionsdrucker wegstellt und an dessen Stelle mit dem Testgerät praktisch arbeitet. In dieser Zeit werden alle Funktionen des Druckers ausprobiert, seine Handhabung bewertet und versucht, Anpassungen an verschiedene Programme zu machen. Außerdem werden alle zur Verfügung stehenden Erweiterungen wie Einzelblatteinzug und verschiedene Schnittstellen durchgetestet.

Fortsetzung Referenzdrucker Seite 70



Bild 1. Unser Druckerspezialist beim Test

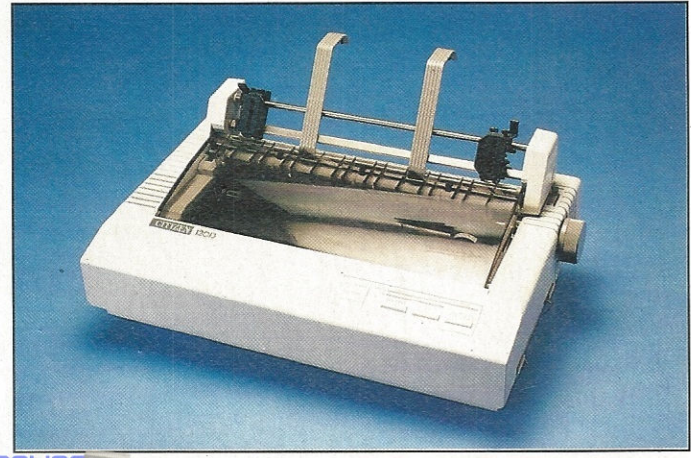


Bild 2. Citizen 120 D — Referenzdrucker der Preisklasse I



Bild 3. Star NL-10 — Referenzdrucker der Preisklasse II

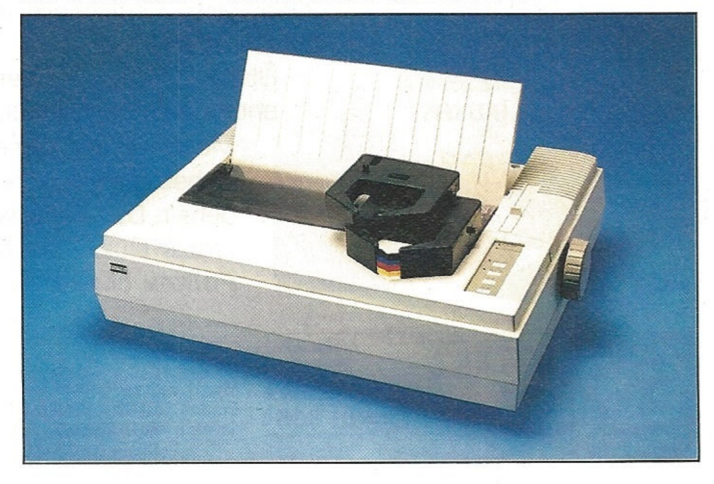
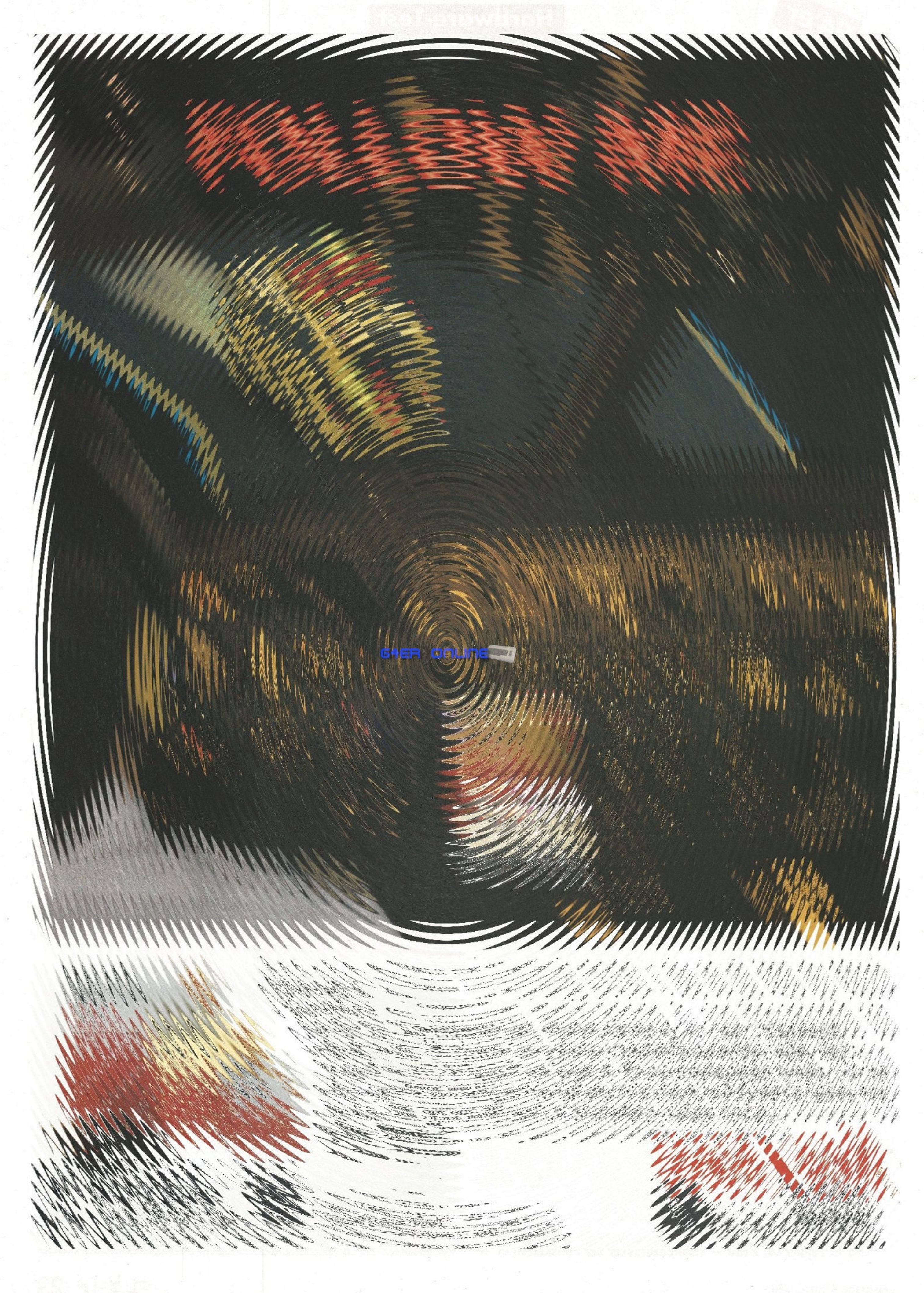


Bild 4. Fujitsu DX 2100 — Referenzdrucker der Preisklasse III ►



Der Computer greift um sich

Lampen blinken, jede Menge Züge fahren auf der Eisenbahnanlage, ohne daß ein Chaos entsteht. Der Roboter arbeitet unermüdlich und Ihr Computer führt die Regie. All dies können Sie nur verwirklichen, indem Sie mit Ihrem Computer messen — steuern — regeln.

Der Wunsch nach Robotern ist fast so alt wie die Menschheit selbst. Schon im antiken Griechenland baute man mechanische Tempeldiener, die bestimmte Bewegungen ausführen konnten — gesteuert durch das Gewicht von Sand, der aus großen Sanduhren herabrieselte. Wie Sie merken, existiert die Problematik »Messen — Steuern — Regeln« wesentlich länger als man annehmen könnte. Die moderne Technik eröffnet uns heute natürlich ganz andere Anwendungsbereiche. Roboter schweißen Autos zusammen, lackieren sie oder wechseln Werkstücke an Maschinen. Dabei sind wir auf Sand für die Steuerung nicht mehr angewiesen. Der Computer übernimmt heute diese wichtige Funktion und sorgt dafür, daß die jeweiligen Aufgaben schnell, ausdauernd und mit enormer Präzision durchgeführt werden. Daher kommt dem Gebiet Messen — Steuern — Regeln heute eine besondere Bedeutung zu. In Ihrem C 64 schlummern übrigens auch dafür enorme Fähigkeiten. Wie man hört, soll der C 64 sogar in einem großen fränkischen Kraftwerk als unermüdlicher Meßdiener arbeiten. Aber wenden wir uns erst einmal den Grundlagen zu.

Regeln leicht gemacht

Sie sitzen wieder einmal vor Ihrem Computer und wollen schnell noch ein Programm fertigstellen. Inzwischen hat der Uhrzeiger aber schon beachtliche Wege zurückgelegt und die in der Nähe stehende Kaffeemaschine leistete einen entscheidenden Beitrag dazu, Sie wachzuhalten. Und schon sind Sie wieder mit dem Problem Messen — Steuern — Regeln konfrontiert, denn in einer Kaffeemaschine wird gemessen, gesteuert und geregelt. Sie werden sich fragen: Was wird in einer Kaffeemaschine denn geregelt? Wir wollen es Ihnen erklären.

Da kalter Kaffee ja geschmacklich nicht das Gelbe vom Ei ist, muß die dunkelbraune Flüssigkeit heiß

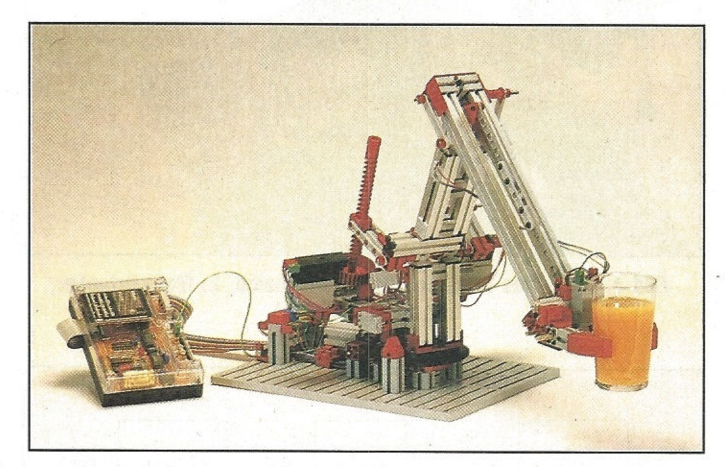


Bild 1. Roboter von Fischertechnik mit Interface für den C 64

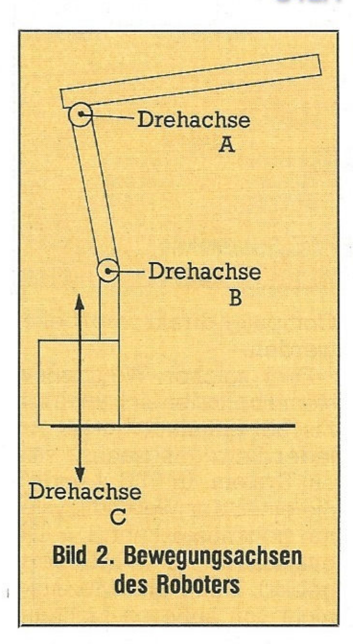


Bild 2. Bewegungsachsen des Roboters

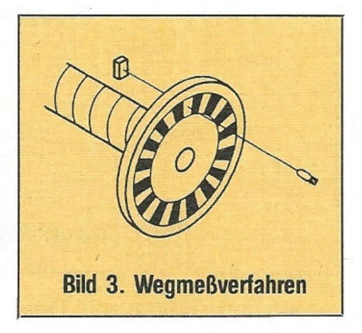


Bild 3. Wegmeßverfahren

gehalten werden. Solange, bis auch der letzte Tropfen aus dem Glasbehälter in die Tasse gegossen wird. Eine eingebaute Heizplatte übernimmt diese Aufgabe. Wenn Sie diese einschalten, haben Sie übrigens schon einen Steuervorgang vollzogen. Aber zurück zum Kaffee, denn der soll nicht kochen und in kürzester Zeit verdampfen, sondern eben nur heiß bleiben. Also muß irgendwann die Heizung abgeschaltet werden (übrigens wieder ein Steuervorgang). Deshalb wird die Temperatur der Heizplatte gemessen. Doch kommen wir nun zu der entscheidenden Verknüpfung vom Messen und Steuern, die für eine Regelung notwendig ist. Die gemessene Temperatur wird mit einem Sollwert (Kaffee kocht fast) verglichen. Ist bei unserer Kaffeemaschine die Solltemperatur erreicht, wird die Heizung abgeschaltet. Die Kaffeekanne nebst ihrem Inhalt ist jedoch permanent bestrebt, Wärme an die Umgebung abzugeben. Das bedeutet, die Temperatur des Kaffees (und damit auch der Heizplatte) sinkt. Es tritt eine Störung ein, der entgegengewirkt werden muß. Deshalb wird die Heizung wieder eingeschaltet, nachdem die Solltemperatur um einige Grade unterschritten ist. Auf diese Weise ist Ihr Kaffee weder zu heiß noch zu kalt.

Einen typischen Regelfall haben Sie hier vor sich, denn die Aufgabe einer Regelung besteht darin, eine bestimmte physikalische Größe (Regelgröße) auf einen vorgegebenen Sollwert zu bringen und dort zu halten. Störungen, die dabei auftreten, werden durch die Regelung ausgeglichen. In dem beschriebenen Fall ist diese Regelgröße die Temperatur — und zwar die Temperatur des Kaffees.

Aber verlassen wir nun das Reich der dunklen Bohnen und wenden uns wieder dem Computer zu.

Jetzt regelt der Computer

Der Computer ist hervorragend dafür geeignet, Regelaufgaben zu übernehmen. Durch die schnellen Arbeitsroutinen kann er in kürzester Abfolge Sollwerte mit Istwerten (Meßwerten) vergleichen und entsprechende Maßnahmen veranlassen.

Unsere moderne Technik hat uns den künstlichen Hilfsarbeiter beschert, den Roboter. Seine Bewegungen werden von Computern gesteuert, überwacht und geregelt. An einem kleinen Bruder der imponierenden Industrieroboter, der auch auf Ihren Schreibtisch paßt, wollen wir Ihnen zeigen, wie sie funktionieren.

In Bild 1 sehen Sie den von Fischertechnik entwickelten Roboter mit Interface für den C 64. Er stellt sich Ihnen auch gleich mit einem Glas in der Hand oder besser gesagt, im Greifer, vor. Sie sehen aber nicht, wie sich der Roboter bewegt — deshalb ein paar Informationen darüber.

Vielen Robotern diente der menschliche Arm als Vorbild. Auch der Roboter von Fischertechnik besteht aus einem Ober- und einem Unterarm. Beide drehen

sich um je eine Achse. Im Bild 2 finden Sie die beiden Achsen — bezeichnet mit Drehachse A und B. Damit der Roboter jeden Punkt seiner Umgebung anfahren kann, muß eine horizontale Drehung ebenfalls möglich sein (Bild 2, Achse C). Man spricht deshalb bei diesem Roboter von einer dreiachsigen Bewegung, für die auch drei Antriebsmotoren notwendig sind. Einmal abgesehen von den mechanischen Grenzen, kann der Roboter so jeden beliebigen Punkt erreichen. Sie brauchen die einzelnen Motoren nur für eine bestimmte Zeit einzuschalten und der Roboter hat die entsprechende Position erreicht.

Doch halt! Was ist, wenn der Roboter eine große Last transportieren soll und die Motoren sich langsamer drehen? Der Zeitplan kommt durcheinander und schon legt der Roboter seine Last an der falschen Stelle ab. Auch bei Schwankungen der Betriebsspannung kann ein Fehler bei der Positionierung entstehen. Dagegen muß dringend etwas unternommen werden. Die Lösung haben Sie sicher schon geahnt. Die Bewegungen des Roboters müssen geregelt werden. Aber jetzt übernimmt der Computer diese Aufgabe.

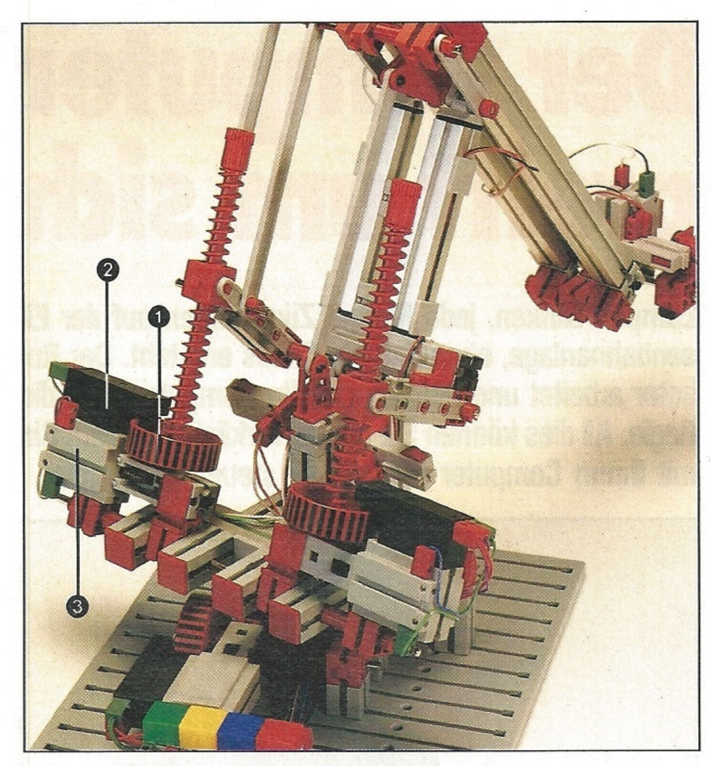


Bild 4. Detail des Roboters (1: Digitalscheibe, 2: Meßwertgeber, 3: Antriebsmotor)

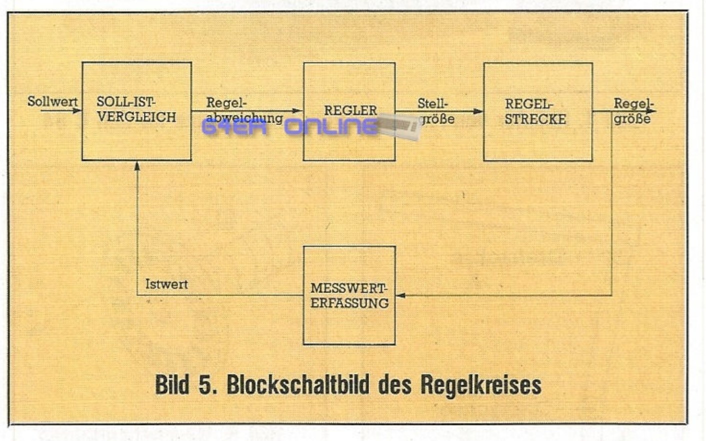


Bild 5. Blockschaltbild des Regelkreises

Roboter im Einsatz

Wie Sie aus dem ersten Beispiel sicher noch wissen, ist ein Meßvorgang für eine Regelung notwendig. Was muß in diesem Fall gemessen werden? Hier sind es die vom Roboterarm zurückgelegten Wege, die zu messen sind. Wenn Sie sich nun Gedanken über ein geeignetes Meßverfahren machen, so sollten Sie beachten, daß der Computer nur digitale Werte verarbeiten kann. Das heißt, der Computer erkennt nur zwei Schaltzustände, nämlich 0 und 1. Eine Weginformation liegt aber normalerweise in analoger Form vor. Dabei ist jeder beliebige Wert möglich wie beispielsweise 3,18 oder 16,3. Eine solche Information müßte erst durch einen Analog-/Digital-Wandler umgeformt werden, wenn der Computer sie verarbeiten soll. Deshalb wurde bei unserem Roboter ein Wegmeßverfahren gewählt, das gleich einen digitalen Meßwert liefert.

In Bild 3 können Sie erkennen, wie so etwas gemacht wird. Die Antriebsspindel ist mit einer sogenannten Digitalscheibe versehen. Auf dieser befindet sich ein Raster aus hellen und dunklen Flächen. Dreht sich nun diese Scheibe, so erzeugt eine optische Abtasteinheit, die aus einer Lichtquelle und einem Fotoelement besteht, eine Impulsfolge. Damit haben wir eine digitale Weginformation, bei der ein Impuls einem bestimmten Weg entspricht. Durch Aufsummieren der Impulse erhalten Sie den zurückgelegten Weg. Diese Information kann vom Computer direkt verarbeitet werden.

Drei solcher Wegmeßsysteme befinden sich auch am Fischertechnik-Roboter. An jeder Bewegungsachse sitzt ein System. In Bild 4 finden Sie eines der Wegmeßsysteme gekennzeichnet (1 = Digitalscheibe, 2 = Meßwertgeber). Damit die Messung nicht von äußeren Lichteinflüssen gestört werden kann, verwendet Fischertechnik ein Infrarot-Meßsystem. Dabei läßt der rote Kunststoff Infrarotlicht durch.

Nachdem der Meßvorgang nun hinreichend besprochen wurde, wollen wir uns der Regelung widmen. Bild 5 zeigt das Blockschaltbild für einen Regelkreis, wie er hier vorhanden ist. Gehen wir davon aus, daß der Roboter eine neue Position anfahren soll. Ein neuer Sollwert wird also vorgegeben. Natürlich unterscheidet sich der Istwert jetzt erheblich von dem Sollwert. Daraus ergibt sich die Regelabweichung. Der Computer reagiert nun als Regeleinrichtung (Regler) und erzeugt einen Stellbefehl für die Regelstrecke. In unserem Fall bedeutet dies, daß der Motor mit der entsprechenden Bewegungsrichtung eingeschaltet wird und eine Positionsänderung des Roboters bewirkt. Die derzeitige Position (Regelgröße) wird nun wiederum gemessen und ein neuer Istwert dem Vergleicher zugeführt. Diese Vorgänge wiederholen sich in sehr kurzen Zeitabständen, bis der Istwert mit dem Sollwert übereinstimmt. Der Regler reagiert dann, indem er den Antriebsmotor ausschaltet. Sollte der Motor beispielsweise nachlaufen, so entsteht wieder eine Regelabweichung, auf die der Regler entsprechend reagiert. Auf diese Weise kann unabhängig von äußeren Störeinflüssen eine genaue Positionierung erreicht werden.

Geregelte Wege

Eine solche Regeleinrichtung existiert für jede der drei Bewegungsachsen. Im Bild 4 sehen Sie auch die drei Meßeinrichtungen. Interessant ist, daß alle drei Regelvorgänge quasi parallel ablaufen können. Durch ein Maschinenprogramm sind die Arbeitsroutinen des Computers entsprechend schnell.

Die notwendigen Steuervorgänge und Istwerterfassungen erfolgen über den Port B des User-Ports.

Dies war nur ein Beispiel dafür, wie sich der C 64 zum Messen — Steuern — Regeln einsetzen läßt. Ebenso können Sie mit Ihrem Computer die Aquarientemperatur regeln, Blumen bewässern oder die Heizungsanlage regeln. Aber wir wollen Ihrer Fantasie keine Grenzen setzen. (kn)

Info: Fischerwerke, Arthur Fischer GmbH & Co. KG, 7244 Tumlingen/Waldachtal

Die Sinne eines Computers

Sensoren sind die Ohren, Augen und Nase eines Computers. Es gibt kaum eine physikalische Größe, die sicher vor ihnen ist. In der heutigen Technik sind sie das A und O der Computersteuerung.

Computer können hören, sehen und riechen! Man ist häufig nur etwas erstaunt, wie eine Handvoll Chips und ein paar Drähte das fertigbringen. Das Zauberwort heißt »Sensoren«. Im folgenden wollen wir Ihnen die wichtigsten vorstellen.

Sensoren sind Bauelemente die durch Einfluß von magnetischen, elektrischen oder mechanischen Größen ihre elektrischen Eigenschaften ändern oder Energien in Ströme umwandeln. Sensoren, die ihre elektrischen Eigenschaften (meist den Widerstand) ändern, werden als passiv bezeichnet. Aktive Sensoren indessen wandeln chemische, elektromagnetische, thermische und mechanische Energie in elektrischen Strom, der sich sofort messen läßt.

Die einfachsten Sensoren sind die Passiven. Sie ändern einfach ihren Gleichstromwiderstand. Ein Sensor den fast jeder kennt, wird zur Längen- und Winkelmessung verwendet: Das Schiebe- oder Drehpotentiometer, auch kurz Poti genannt. Am einfachsten läßt sich das Schiebepotentiometer (Schieberegler) erklären.

Potentiometer

Auf einer Bahn befindet sich eine Schicht aus Kohle über die ein Schleifer hin und her geschoben werden kann (Bild 1). Mit einer Hilfsstromquelle kann die Widerstandsänderung in eine Spannungsänderung umgewandelt werden. Schiebepotentiometer gibt es in Längen bis zu einem Meter. Meist werden sie für Wegmessungen an Maschinen verwendet. Drehpotentiometer werden für Winkelmessungen hergenommen.

Wenn Elektronik Ihr Hobby ist, wissen Sie bestimmt, daß normale Kohleschichtpotis alles andere als genau sind. Aus diesem Grund findet man in Meßpotentiometer auch keine Kohleschicht, sondern eine Schicht aus speziellem Leitplastik, das wesentlich abriebfester ist. Die Widerstandsänderung durch Abrieb kann dadurch fast vernachlässigt werden.

Durch Widerstandsänderungen können nicht nur leicht Wegstrecken gemessen werden, sondern auch Temperaturen. Möglich wird das mit einem Metallthermometer (Bild 2). Wie Sie vielleicht noch aus Ihrem Physikunterricht wissen, ändern Metalle ihre Leitfähigkeit mit der Temperatur. Je größer die Temperatur, um so schlechter leiten sie. Ein Metallthermometer läßt sich dadurch leicht beschreiben. Man wickelt auf ein Glasröhrchen eine Lage Platindraht und schon hat man einen Thermofühler für einen Temperaturbereich von -220°C bis +1000°C. Aber ganz so einfach ist nun auch nicht. Denn die Zuleitungen zur Platinwendel ändern auch ihren Widerstand. Zusätzlich muß der Meßstrom durch die Platinwendel sehr klein sein, damit der Platindraht nicht zum »Heizdraht« wird. Wenn nämlich der fließende Strom den Platindraht geringfügig erwärmt, wird das Meßergebnis total verfälscht. Aus diesen Gründen hat man sich etwas anderes überlegt, das in der Praxis einfacher handzuhaben ist: Das Halbleiterthermometer.

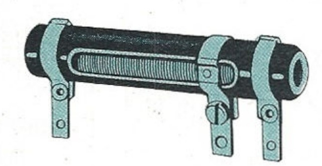


Bild 1. Ein Schiebepoti hat eine Drahtbahn, auf der ein Schleifer hin- und hergeschoben werden kann.

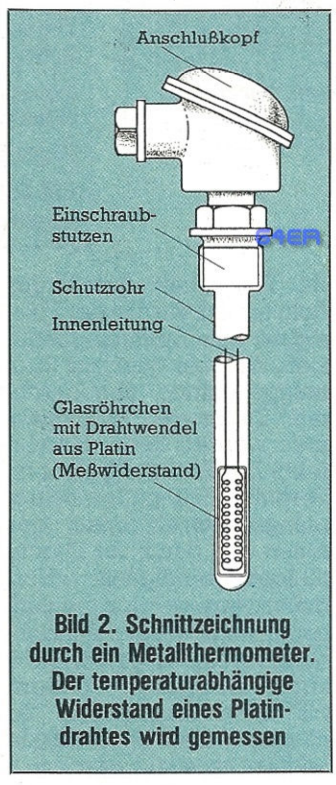


Bild 2. Schnittzeichnung durch ein Metallthermometer. Der temperaturabhängige Widerstand eines Platindrahtes wird gemessen

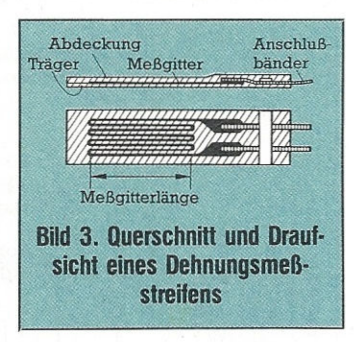


Bild 3. Querschnitt und Draufsicht eines Dehnungsmeßstreifens

Es gibt zwei verschiedene Arten von Halbleiterthermometern: Kalt- und Heißleiter. Kaltleiter haben einen positiven Termperaturkoeffizieten (PTC, Widerstand steigt mit zunehmender Temperatur), Heißleiter einen negativen (NTC, Widerstand sinkt mit steigender Temperatur). Der Vorteil der Halbleiterthermometer ist ihr Temperaturbeiwert, also die Größe der Widerstandsänderung pro Grad Celsius. Die Widerstandsänderungen bei Heiß- und Kaltleitern sind wesentlich größer als bei den Metallthermometern. Eine Temperaturdifferenz ist deshalb leichter zu messen. Außerdem ist die Masse kleiner als die der Metallthermometer. Das bedeutet, daß Temperaturänderungen schnell gemessen werden können. Das kleine Gehäuse der Halbleiterthermometer nimmt schnell die Temperatur des Meßobjektes an, ohne dessen Temperatur stark zu beeinflussen. Der Nachteil ist, daß die Widerstandsänderung nicht linear zur Temperaturänderung ist. Bei den aktiven Sensoren werden Sie einen Thermofühler kennenlernen, der die Vorteile des Metall- und des Halbleiterthermometers in sich vereinigt.

Materialprüfung

In der Materialprüfung werden häufig Belastungen an statischen Objekten wie Brückenträgern oder anderen Stahlkonstruktionen gemessen. In der Regel gibt es hier Zug- und Druckkräfte zu messen, das Einsatzgebiet von Dehnungsmeßstreifen (DMS). Die Wirkungsweise eines DMS ist schnell erklärt: Durch Dehnung eines Drahtes wird dessen Querschnitt verkleinert, was eine Widerstandserhöhung bewirkt. Längenänderungen von 0,1 bis 10 Mikrometer können damit gemesssen werden. Bild 3 zeigt den schematischen Aufbau eines Folien-Dehnungsmeßstreifens. Ähnlich der Herstellung von Platinen, wird ein metallisches Meßraster auf eine Folie aufgebracht, die leicht auf das Meßobjekt geklebt werden kann. Der Querschnitt der in »Schlangenlinien« verlegten Meßleitung ist in Längsrichtung kleiner als in den Kurven. Dadurch wird der DMS unempfindlich gegen etwaige Dehnungen in Querrichtung. Zur Messung von Dehnungen in verschiedene Richtungen gibt es spezielle DMS, deren Meßgitter in verschiedene Richtungen verlaufen (Bild 4). Als Werkstoff dient meist eine Konstantan-Nickel- oder Chrom-Nickel-Legierung. Ein Dehnungsmeßstreifen kann auch wieder gestaucht werden, wenn er gestreckt wurde.

DMS finden sich auch in sogenannten Kraftmeßdosen (Bild 5), die an Pressen, Walzen und in elektroni-

schen Waagen zu finden sind. Kräfte bis zu 1000 KN können damit gemessen werden.

Es gibt noch eine andere Art von Kraftmeßdosen, die kapazitive Effekte ausnutzen. Bild 6 zeigt eine solche kapazitive Druckmeßdose. Durch Eindrücken der Membran, wird die Kapazität der Dose verändert. Vergleichen läßt sich das mit einem Kondensator, dessen Plattenabstand variabel ist. Gemessen wird die Kapazitätsänderung mit einer Wechselstrommeßbrücke, in der die Druckmeßdose Teil eines Schwingkreises ist. Ändert sich die Kapazität, wird der Schwingkreis verstimmt. Indem man den Schwingkreis mit einem RC-Glied wieder auf die Resonanzfrequenz abstimmt, kann man die Kapazitätsänderung bestimmen.

Auch kapazitive Füllhöhenmesser für elektrisch nicht leitende Flüssigkeiten funktionieren nach diesem Prinzip (Bild 7). Behälterwand und Mittelelektrode bilden einen Kondensator. Die Flüssigkeit ist das Dielektrikum (=Medium zwischen Kondensatorplatten). Da der kapazitive Widerstand eines Kondensators auch von der Art des Dielektrikums abhängt, kann die Füllhöhe relativ leicht bestimmt werden. Ist der Behälter voll, ist über die ganze Höhe die Flüssigkeit das Dielektrikum. Ist der Behälter nur halbvoll, ist zur Hälfte die Flüssigkeit und zur Hälfte Luft das Dielektrikum. Der volle Behälter hat dadurch eine kleineren kapazitiven Widerstand als der halbvolle, da Luft nur ein schlechtes Dielektrikum ist. Müssen leitende Flüssigkeiten gemessen werden, isoliert man einfach die Elektrode und verwendet die Flüssigkeit selbst als Kondensatorplatte.

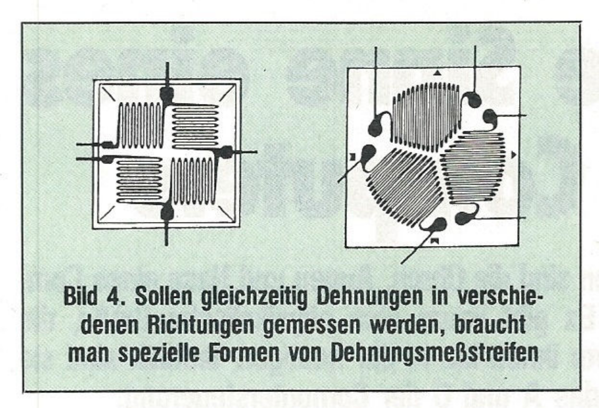


Bild 4. Sollen gleichzeitig Dehnungen in verschiedenen Richtungen gemessen werden, braucht man spezielle Formen von Dehnungsmeßstreifen

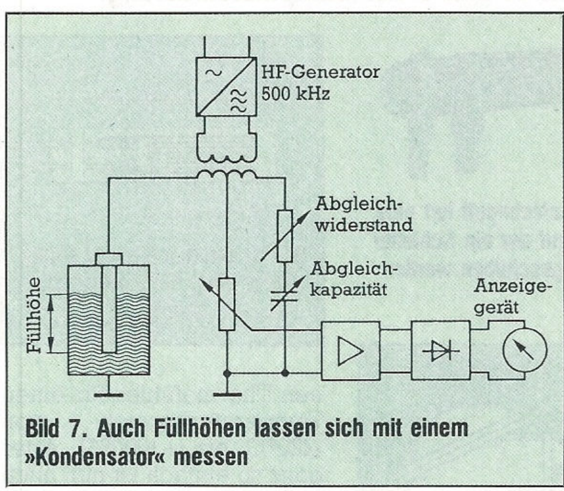


Bild 7. Auch Füllhöhen lassen sich mit einem »Kondensator« messen

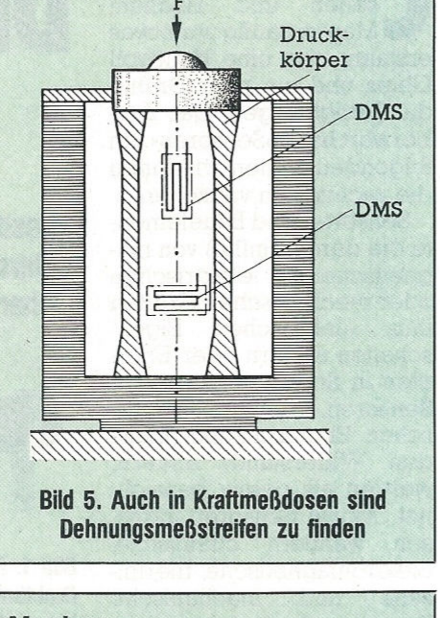


Bild 5. Auch in Kraftmeßdosen sind Dehnungsmeßstreifen zu finden

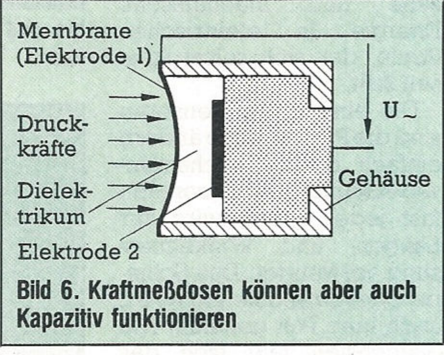


Bild 6. Kraftmeßdosen können aber auch Kapazitiv funktionieren

Mengenmessung

Hat ihr Wagen eine L-Jetronic als Einspritzanlage, dann wissen Sie auch wie wichtig es sein kann Durchflußmengen zu messen. L-Jetronic steht frei übersetzt für Luftmengen-Messung. Die Wirkungsweise ist wieder sehr leicht zu verstehen. Durch Anblasen mit Luft kann man einen Gegenstand schneller abkühlen als ohne Anblasen. Das ist der ganze Trick. Im Ansaugkanal einer L-Jetronic befindet sich eine Hitzdrahtsonde (Bild 8), an der die Ansaugluft vorbeiströmt. Die Sonde wird nun über eine konstante Stromquelle aufgeheizt. Saugt nun der Motor Luft an, wird die Sonde gekühlt und verändert dadurch ihren Widerstand. Sie wissen ja: kalte Metalle haben einen kleineren Widerstand als heiße. Je mehr Luft angesaugt wird, um so kleiner wird der Widerstand der Sonde. Aus dem Widerstand der Sonde läßt sich dann die angesaugte Luftmenge bestimmen und die benötigte Kraftstoffmenge errechnen.

Die Hitzdrahtsonde kann aus einem dünnen Wolframdraht, der an den Enden auf einen Nickelfühlerhalter geschweißt wird, bestehen.

Sehr genau können Durchflußmengen von leitenden Flüssigkeiten auch induktiv gemessen werden (Bild 9). Solche magnetischen Durchflußsensoren bestehen aus einem isolierten Rohr, das sich in einem Magneten befindet. Wenn nun die leitende Flüssigkeit durch das Rohr strömt, wird in der Flüssigkeit eine Spannung induziert, die von quer zur Strömungsrichtung angebrachten Elektroden abgenommen werden kann (Hall-Effekt). Die Spannung steht im Verhältnis zur mittleren Strömungsgeschwindigkeit der Ionen und damit zur Durchflußgeschwindigkeit. Normalerweise werden die Sensoren, nicht wie in der Zeichnung vereinfacht, mit Gleichstrom betrieben, sondern mit Wechselspannung. So können die Spannungen leichter verstärkt werden.

Längenmessung

Passive Sensoren funktionieren nicht nur durch Veränderung von Widerständen. Auch magnetische Effekte können zur Meßdatenerfassung eingesetzt werden. Ein noch recht einfaches Beispiel dazu ist der Tauchankersensor (Bilder 10 und 11), der für Wegmessungen von 50 bis 1500 mm eingesetzt wird. Die Hobbyelektroniker unter Ihnen werden schon anhand des Schaltbildes die Funktion erkennen: Ein Rundstab aus Weicheisen wird in ein Rohr mit einer Referenz- und einer Meßwicklung geschoben, wodurch sich natürlich die Induktivität (magnetische Leitfähigkeit) des Rohres verändert. Wird an die Referenzspule eine Wechselspannung angelegt, induziert diese in der Meßwicklung eine Spannung, deren Höhe davon abhängt, wie tief das Rundeisen in der Spule steckt. Häufig verwendet man Tauchankersensoren für Füllstandsanzeigen.

Aktive Thermofühler

Metallthermometer, Heiß- und Kaltleiter haben wie schon beschrieben Nachteile, die es beim Thermoelement nicht gibt. Thermoelemente arbeiten sehr genau und linear über einen großen Temperaturbereich hinweg. Diese Sensoren nutzen einen eigenartigen Effekt aus. Verschweißt man beispielsweise einen Konstantan- mit einem Kupferdraht,



Bild 8. Die Hitzdrahtsonde dient zur Messung von Strömungsgeschwindigkeiten, aus denen leicht eine Durchflußmenge ermittelt werden kann

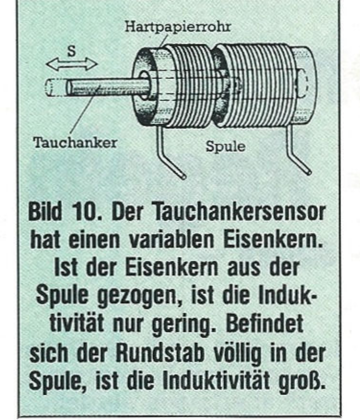


Bild 10. Der Tauchankersensor hat einen variablen Eisenkern. Ist der Eisenkern aus der Spule gezogen, ist die Induktivität nur gering. Befindet sich der Rundstab völlig in der Spule, ist die Induktivität groß.

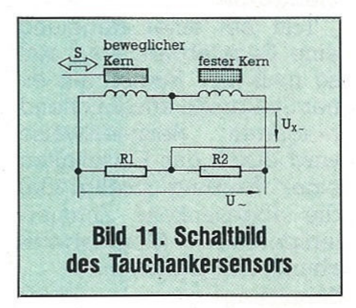


Bild 11. Schaltbild des Tauchankersensors

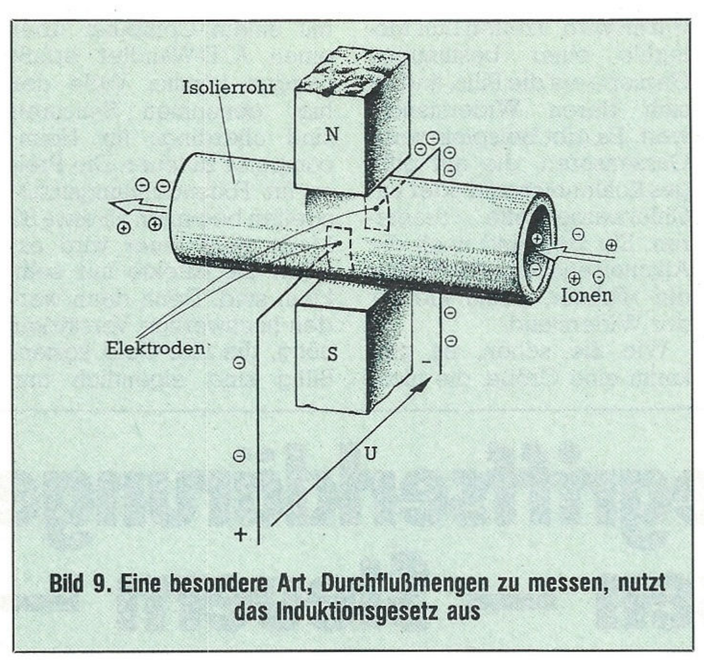


Bild 9. Eine besondere Art, Durchflußmengen zu messen, nutzt das Induktionsgesetz aus

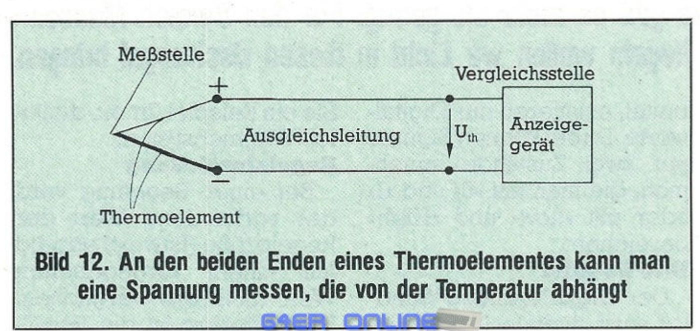


Bild 12. An den beiden Enden eines Thermoelementes kann man eine Spannung messen, die von der Temperatur abhängt



Bild 13. Thermoelemente werden normalerweise in ein Gehäuse eingebaut

so kann man eine Spannung an den Drahtenden messen (Bilder 12 und 13). Die Spannung ist proportional zur Temperatur der Verbindungsstelle. Man mißt normalerweise die Spannungsdifferenz zwischen zwei Thermoelementen, von denen eines auf eine Referenztemperatur gebracht wird und das andere als Meßfühler dient. Hat das Meßelement die Temperatur des Referenzelementes, kompensieren sich die Thermospannungen gegenseitig. Erst bei einem Temperaturunterschied läßt sich eine Thermospannung messen. So läßt sich das Meßelement leicht eichen.

Strom durch Druck

Kräfte können nicht nur durch Druckdosen, sondern aktiv auch durch piezoelektrische Sensoren gemessen werden. Solche Sensoren erzeugen bei Belastung durch Druck-, Schub- und Zugkräfte eine elektrische Spannung. Denn bei Veränderung der Form tritt an den Kristallseiten die parallel zur Kraftrichtung sind, eine Ladungstrennung auf. Auf der einen Seite sind Elektronen im Überschuß, auf der anderen herrscht Elektronenmangel.

Bestimmt kennen Sie diesen Effekt von einigen Feuerzeugen her. Mit dem Taster drücken Sie den Piezokristall so lange zusammen bis eine »Knackfroschmechanik« den Druck schlagartig vom Kristall wegnimmt. Der Kristall federt in seine Ruhelage zurück und erzeugt dabei einen kurzen Stromimpuls. Sichtbar am Funken, der zwischen den Zündelektroden überspringt. Umgekehrt funktioniert ein Piezolautsprecher. Legt man an einen Piezokristall eine Wechselspannung an, tritt eine periodische Verformung auf, die hörbar sein kann.

In Sensoren werden meistens Blei-Zirkonat-Titanat-Kristalle verwendet. Diese sind wesentlich effektiver als die bekannten Quarz-Kristalle. Der Wirkungsgrad liegt bei etwa 50 Prozent.

Mit piezoelektrischen Aufnehmern können allerdings nur dynamische Systeme gemessen werden. Denn sobald eine konstante Kraft auf den Kristall einwirkt, tritt auch keine Ladungstrennung im Kristall mehr auf.

Geschwindigkeits- und Drehzahlmessung

Geschwindigkeiten und Drehzahlen werden mit Tachogeneratoren gemessen. Prinzipiell ist ein Tachogenerator nichts anderes als ein Stromgenerator, der eine um so größere Spannung erzeugt, je schneller er sich dreht. Aus der Höhe der Spannung läßt sich leicht die Drehzahl eines Motors ermitteln.

Beschleunigungen können nach verschiedenen Verfahren gemessen werden. Alle haben jedoch eines gemeinsam: es wird die Kraft gemessen, welche eine schwingend aufgehängte »träge Masse« auf den Aufhängungspunkt ausübt. Die Kraftmessung erfolgt meist piezoelektisch oder über eine Druckdose.

Was mit einem Heimcomputer einfach gemessen werden kann, ist neben Temperaturen auch Licht. Das bekannteste und älteste Bauelement, das auf Licht reagiert, ist der Fotowiderstand. Er besteht aus einem Halbleitermaterial, dessen Elektronen bei Lichteinfall frei beweglich werden und somit steigt die Leitfähigkeit. Durch geeignete Wahl des Halbleitermaterials kann man solche Fotowiderstände für verschiedene Spektralbereiche empfindlich machen (sensibilisieren). Die Widerstandsänderung von LDRs (Light Dependend Resistor) ist annähernd logarithmisch zur einfallenden Lichtstärke. Der Nachteil der LDRs sind ihre lange Ansprechzeit. Es dauert eine Zeit lang, bis eine Lichtänderung eine Widerstandsänderung hervorruft. Nicht so bei der Fotodiode, die sofort anspricht. Momentan ist sie der schnellste lichtempfindliche Schalter. Die Reaktionszeiten bewegen sich im Zeitraum von einer Mikrosekunde. Ein Fotowiderstand braucht immerhin bis zu 20 Millisekunden, um sich auf eine Lichtänderung einzustellen. Eine Fotodiode wird in Sperrichtung betrieben und hat ihre maximale Empfindlichkeit im dunkelroten Licht. Das dritte Bauteil, das auf Licht reagiert, ist der Fototransistor. Er verbindet die Vorteile von Fotowiderstand und Diode. Die Schaltzeiten bewegen sich in ähnlichen Bereichen wie die der Fotodiode und die spektrale Empfindlichkeit ist über einen weiten Bereich gestreckt.

Es gibt auch spezielle Gassensoren, die auf eine Vielzahl von Gasen ansprechen, allerdings braucht jede Gas-

Riechen kann er auch noch!

sorte einen eigenen Sensor. Diese Sensoren bestehen aus einer Halbleiterpille, die durch eine Heizwendel erwärmt wird. Treffen nun Moleküle einer bestimmten Gassorte auf die Pille, ändert sich deren Widerstandswert. Es gibt beispielsweise Gassensoren, die auf giftiges Kohlenmonoxid oder auf Chlorwasserstoffe reagieren. Ein Gag sind noch die Alkoholsensoren. Je stärker die »Fahne«, desto kleiner der Widerstand.

Wie Sie sehen, es gibt kaum eine Größe, die nicht mit einem Computer über einen A/D-Wandler erfaßt werden könnte. Viele der hier genannten Sensoren sind allerdings für Heimcomputer zu teuer. Die Preise für Foliendehnungsmeßstreifen beginnen ab etwa 30 Mark. Sehr teuer wird es, wenn die Effekte nur sehr klein sind. Denn dann werden hochwertige Verstärker nötig, die viel Geld kosten. Billig sind eigentlich nur Heißleiter, NTCs, und Fotowiderstände, Fotodioden und Fototransistoren. Eventuell noch Thermoelemente. Zum Teil gibt es diese Bauteile schon für weniger als eine Mark in Elektronikgeschäften. (hm)

Info: Als Quelle zu diesem Thema diente das Buch »Fachkunde für Informationselektronik«, das im Europa-Lehrmittel-Verlag erschienen ist. Der Preis beträgt 45 Mark.

Begriffserklärungen zu Messen — Steuern — Regeln

Fachausdrücke gibt es mehr als genug. Für den Bereich Messen — Steuern — Regeln wollen wir Licht in diesen Dschungel bringen.

Analogwert
Analoge Größen können jeden beliebigen Wert annehmen. Wenn sie beispielsweise ein Gummieband auseinanderziehen, können Sie damit unendlich viele Längenwerte (Analogwerte) darstellen. Nahezu alle in der Natur vorkommenden Größen, wie Temperatur, Gewicht oder Druck, sind analoge Größen.

A/D-Wandler
Der Analog-/Digital-Wandler setzt analoge Spannungswerte in digitale Signale um. Da der Computer nur digitale Informationen verarbeiten kann, muß beispielsweise ein analoger Meßwert, wie eine Temperatur von 21,3°C, in eine für den Computer lesbare Impulsfolge übersetzt werden.

Bit
Bit ist die Abkürzung für »Binary Digit«. Ein Bit ist die kleinste Informationseinheit in der digitalen Datenverarbeitung. Mit einem Bit können im binären Zahlensystem nur die Dezimalzahlen 0 und 1 dargestellt werden.

Byte
Ein Byte ist eine Informationseinheit von acht Bit. Damit können Dezimalzahlen von 0 bis 255 dargestellt werden. Die nächst größere Einheit ist KByte (= 2^8 = 1024 Byte).

Digitalwert
Im binären Zahlensystem, mit dem jeder Computer arbeitet, existieren nur Digitalwerte. Dabei können Signale nur zwei Zustände annehmen, die man mit »0« und »1« oder mit »Low« und »High« bezeichnet.

D/A-Wandler
Der Digital-/Analog-Wandler setzt digitale Informationen in analoge Signale um. Wollen Sie beispielsweise die Helligkeit einer Lampe mit dem Computer steuern, so muß die vom Computer kommende Impulsfolge vom D/A-Wandler in ein analoges Steuersignal werden.

Interface
Als Interface bezeichnet man eine Anpassungsschaltung. Mit dieser ist der Austausch von Daten- oder Steuerinformationen zwischen zwei Geräten (Schnittstellem) möglich. Soll Ihr Computer eine Eisenbahnanlage steuern, so ist ein Interface dazu notwendig. Viele Drucker können auch nur über ein Interface am C 64 betrieben werden.

Meßwertaufnehmer
Der Meßwertaufnehmer setzt eine zu messende physikalische Größe in ein elektrisches Signal um. Dabei ergibt sich ein analoger oder digitaler Meßwert. Meßwertaufnehmer werden auch als »Sensoren« bezeichnet. Eine Aufzählung der verschiedensten Sensoren finden Sie in dem Artikel »Die Stimme Ihres Computers«. Im Bild 4 auf Seite 26 sehen Sie ein Beispiel für ein digitales Wegmeßsystem.

Regelabweichung
Bei einer Regelung wird der vorhandene Wert der Regelgröße (Istwert) ständig mit einem vorgegebenen Wert (Sollwert) verglichen. Die Differenz ist die Regelabweichung. Soll beispielsweise eine Temperatur von 23°C erreicht werden und 18°C werden gemessen, so ergibt sich eine Regelabweichung von 5°C.

Regelgröße
Als Regelgröße wird die zu regelnde physikalische Größe bezeichnet. Dies kann beispielsweise eine Temperatur, eine Strecke oder eine elektrische Spannung sein.

Regelung
Eine Regelung hat die Aufgabe, eine physikalische Größe auf einen bestimmten Wert zu bringen und dort zu halten. So müssen beispielsweise in einem tropischen Gewächshaus Temperatur und Luftfeuchtigkeit geregelt werden.

Roboter
Ein Roboter ist ein elektronisch gesteuertes Gerät, das manuelle Funktionen des Menschen ausführen kann.

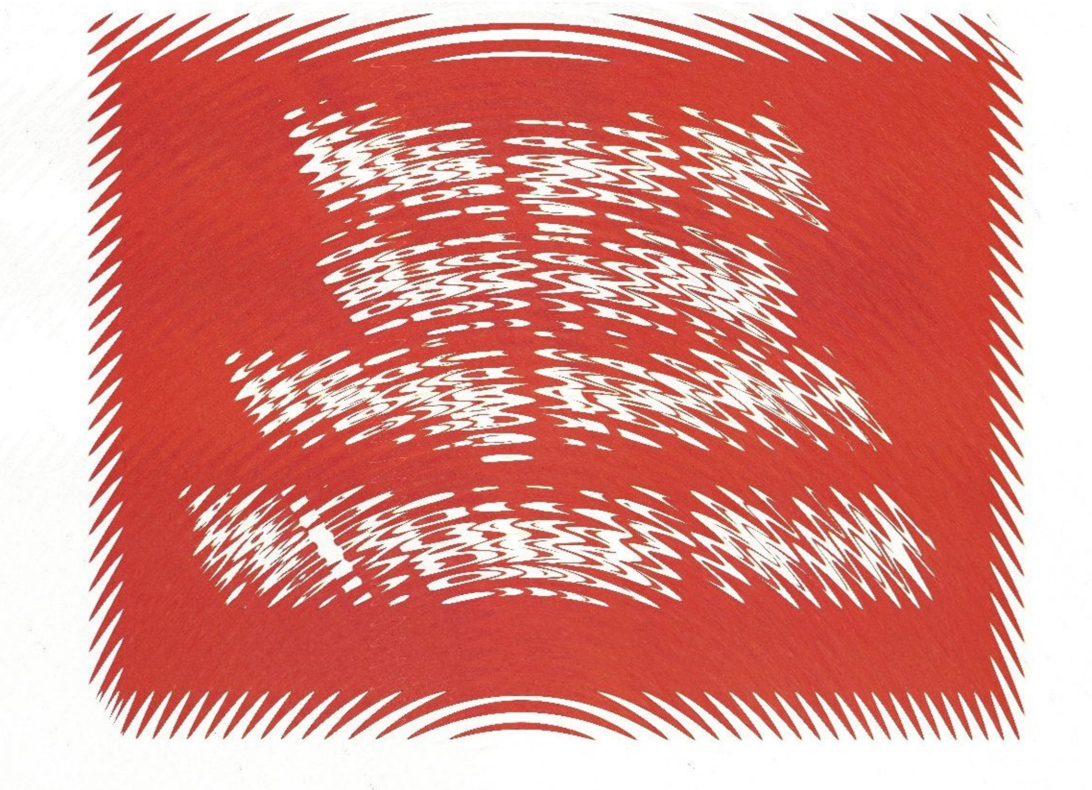
Schnittstelle
Als Schnittstelle bezeichnet man den Anschluß, über den zwischen dem Computer und einem weiteren Gerät Daten übertragen werden können. Ein Beispiel dafür ist die serielle Schnittstelle am C 64, über die Daten von und zum Diskettenlaufwerk übertragen werden.

Stellgröße
Tritt bei einer Regelung eine Regelabweichung auf, so muß der Regler mit einer Stellgröße ausgleichend reagieren. Beispielsweise wird durch das Einschalten einer Heizung (Stellgröße) die abgesunkene Zimmertemperatur (Regelabweichung) ausgeglichen.

Steuerung
Eine Steuerung besteht aus Steuereinrichtung und Stellglied. Die einfachste elektrische Steuereinrichtung ist ein Schalter, mit dem beispielsweise ein Antriebsmotor (Stellglied) ein- und ausgeschaltet wird.

Störgröße
Bei Regelungen ruft die Störgröße eine Änderung der Regelgröße hervor. Aufgabe der Regelung ist es, dieser Störung entgegenzuwirken. Öffnen Sie beispielsweise im Winter das Wohnzimmerfenster, so haben Sie damit eine Störgröße für die geregelte Heizanlage. Die Zimmertemperatur (Regelgröße) sinkt ab und die Heizanlage versucht dies auszugleichen, indem stärker geheizt wird.

User-Port
Der User-Port ist eine Schnittstelle des C 64, über die eine frei programmierbare Ein- und Ausgabe möglich ist. (kn)



Der Computer im Physikunterricht

In den letzten Jahren haben sich Computer in fast allen Bereichen der Naturwissenschaften durchgesetzt. So kommen auch Schulen nicht umhin, sich mit diesem Thema auseinanderzusetzen. Was sich nun mit dem Computer am Beispiel Physikunterricht alles anstellen läßt, wollen wir uns hier etwas genauer anschauen.

Das Computer-System ist an sich, wenn man es genauer betrachtet, nur ein Grundsystem, das aus der Zentraleinheit (Computer mit Tastatur), Monitor oder Fernsehgerät, zumindest einer Datasette oder Diskettenlaufwerk zur Datenspeicherung und eventuell einem Drucker besteht. Aber schon mit diesem Grundsystem läßt sich im Physikunterricht eine Menge anfangen. Nehmen wir zum Beispiel an, Sie haben in einem Versuch, im Rahmen einer Meßreihe, Werte zusammengestellt und wollen diese Werte anhand einer linearen Regression grafisch auswerten. Jeder, der das schon einmal zu »Fuß«, also mit Hilfe eines Taschenrechners probiert hat, kennt die Arbeit, die dahinter steckt. Mit einem Computer ist das anders. Ist die lineare Regression und das entsprechende Unterprogramm zur grafischen Darstellung beliebiger Kurven einmal programmiert, müssen nur noch die Ergebnisse der Meßreihe eingegeben werden. Innerhalb von Sekunden ist dann die Grafik fertig. Dadurch wird die Zeit, die einer Lehrkraft zur Durchführung eines Versuchs zur Verfügung steht, verkürzt. Mit anderen Worten: In den ohnehin recht kurzen Unterrichtsstunden läßt sich mit Hilfe eines Computers effizienter arbeiten. Die gewonnene Zeit kann zum Beispiel dazu dienen, das erlernte Wissen zu vertiefen. Natürlich sollte folgendes beachtet werden: Je intensiver Computer eingesetzt werden, um stupide Rechenarbeit zu erledigen, desto schwerer fällt es den Schülern, Arbeiten dieser Art im Kopf oder mit Hilfe des Taschenrechners auszuführen. Daher ist unbedingt darauf zu achten, daß bestimmte Rechenvorschriften von den Schülern beherrscht werden, bevor der Computer eingesetzt wird.

Computer lassen sich aber nicht nur dazu verwenden, physikalische Messungen auszuwerten. Viel interessanter ist es, die Computer selbst messen zu lassen. Dazu existieren beim C 64 zwei Schnittstellen, von denen sich eine, der User-Port, frei programmieren läßt. Durch ihn stehen dem Anwender acht Leitungen zur Datenein-/-ausgabe zur Verfügung. Jede dieser Datenleitungen läßt sich über ein Steuerregister als Ausgabe- oder Eingabeleitung programmieren.

Dies soll ein Beispiel veranschaulichen, das in jedem Physikunterricht vorkommt, nämlich die Bestimmung der Erdbeschleunigung.

Die Ausgangsgleichung ist »$s=1/2\ a\ t^2$«. Aufgelöst nach »a« für Beschleunigung, wird aus der Gleichung »$a=2s/t^2$«. Die unbekannten Größen sind »s« (für Strecke oder Weg) und »t« (für die Zeit). So viel zu den physikalischen Gleichungen. Der Versuchsaufbau ist denkbar einfach: Man nehme eine Stahlkugel und lasse sie fallen, und zwar durch zwei Lichtschranken, die übereinander angeordnet sind. Die Strecke »s« entspricht nun dem Abstand der beiden Lichtschranken. Was jetzt noch gemessen werden muß, ist die Zeit, die die Stahlkugel benötigt, um den Weg zwischen den Lichtschranken zurückzulegen. Genau das übernimmt der Computer. Die beiden Lichtschranken werden über ein geeignetes Interface an den User-Port angeschlossen. Passiert die Stahlkugel die erste Lichtschranke, muß ein Zähler im C 64 gestartet werden, der dann so lange zählt, bis die Stahlkugel an der zweiten Lichtschranke einen Impuls auslöst. Der entsprechende Zählerstand entspricht der gesuchten Zeit und muß nur noch in Sekunden umgerechnet werden. Eine solche Zeitmessung wäre natürlich genauso gut mit einem Speicheroszilloskop möglich. Jedoch sind Geräte dieser Art extrem teuer und schwer zu bedienen, so daß sie für Versuche, die die Schüler selbst durchführen, ungeeignet sind. Computer sind zwar auch teuer, jedoch haben sie einen Vorteil. Man kann sie so programmieren, daß von den Schülern keine Fehler gemacht werden können.

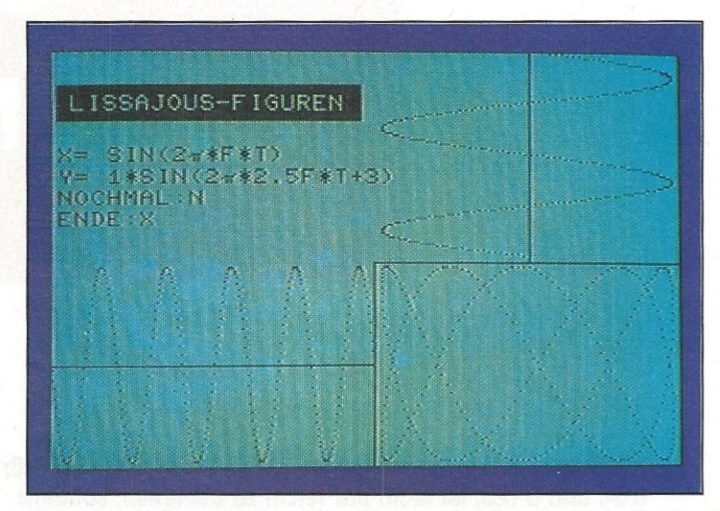


Bild 1. Auch die Darstellung einer Lissajousen-Figur (rechts unten im Bild) ist für den C 64 kein Problem

Mit Hilfe einer kleinen Hardware-Erweiterung, die an den User-Port angeschlossen wird, läßt sich der C 64 zu einem vollwertigen Speicheroszilloskop umfunktionieren. Er hat sogar gegenüber herkömmlichen Geräten einige Vorteile.

Der Computer als Speicheroszilloskop

Aber auch die Nachteile sollen nicht verschwiegen werden.

Vorteile: Die Größe des Bildschirms ist nicht auf 10 mal 10 cm begrenzt, wie das bei »normalen« Speicheroszilloskopen der Fall ist, sondern hängt einzig und allein von der Größe des angeschlossenen Monitors ab. Verwendet man ein Fernsehgerät, kann jeder Schüler Messungen von seinem Platz aus verfolgen.

Die Meßergebnisse lassen sich auf Diskette beziehungsweise Kassette speichern und sind somit jederzeit verfügbar. Aber ein noch entscheidender Vorteil ist der Preis. Gute, herkömmliche Speicheroszilloskope sind nicht unter 2000 Mark zu haben. Der Hardwarezusatz

Wenn vor 10 Jahren jemand behauptet hätte, daß heute, im Jahr 1986, Computer für schulische Zwecke eingesetzt werden, hätte man darüber gelächelt. Die Mikroelektronik hat sich aber in den letzten Jahren so rasant entwickelt, daß Computer heute genauso in den Unterricht gehören wie Deutsch oder Mathematik. Wir zeigen Ihnen hier, was sich mit Computern im Unterricht anstellen läßt.

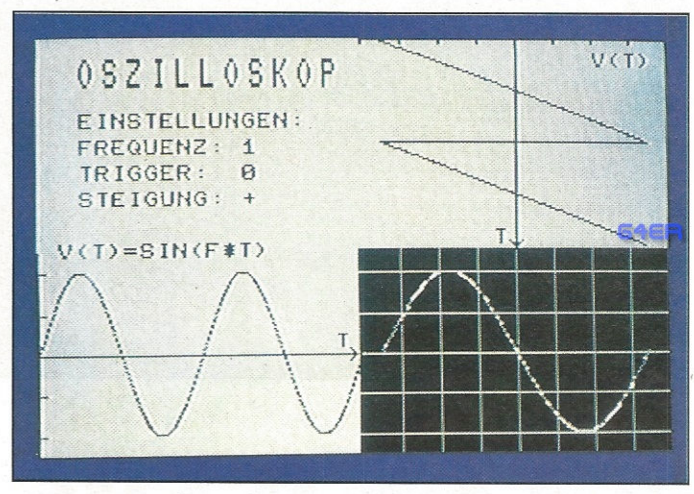


Bild 2. Der C 64 als Oszilloskop. Die Signale am x- und y-Eingang und das daraus resultierende Signal werden gleichzeitig auf dem Bildschirm dargestellt.

und das entsprechende Softwarepaket, das den C 64 zum Speicheroszilloskop umfunktioniert, kostet unter 300 Mark.

Nachteile: Ein normales Speicheroszilloskop ist schnell. Das heißt es lassen sich Frequenzen im Megahertz-Bereich messen beziehungsweise speichern. Da der schon erwähnte Hardwarezusatz für den Computer jedoch einen Analog-/Digital-Wandler enthalten muß, ist die zu messende Frequenz auf einige Kilohertz begrenzt. Ein weiterer Nachteil ist die Tatsache, daß es sich bei den eingebauten A-D-Wandlern um 8-Bit-Wandler handelt. Mit ihnen lassen sich Amplituden nur in 1/256stel-Schritten messen. Da der C 64 aber eine vertikale Auflösung von maximal 200 Punkten hat, dürfte sich dieser Nachteil kaum bemerkbar machen.

Ein Beispiel soll den Einsatz des C 64 als Speicheroszilloskop demonstrieren.

Bei dem Beispiel handelt es sich um ein Thema aus der Schwingungslehre. Es sollen zwei unterschiedliche Schwingungen miteinander verglichen werden, und zwar in bezug auf Frequenz und Phase. Ganz hervorragend eignen sich dazu die Lissajousen-Figuren (Bild 1). Dabei wird der Elektronenstrahl von einer Schwingung in die y- und von der anderen in die x-Richtung abgelenkt. Ist die Frequenz und die Phase beider Schwingungen identisch, so wird das durch einen Kreis auf dem Monitor angezeigt.

Die Lissajousen-Figuren eignen sich außerdem noch dazu, das Funktionsprinzip eines Oszilloskops zu erklären. Wird der Elektronenstrahl nur in x-Richtung abgelenkt, und zwar mit Hilfe eines sägezahnförmigen Signals, so wandert der Leuchtpunkt auf dem Monitor beziehungsweise Fernsehgerät in Abhängigkeit von der Frequenz des Signals entsprechend schnell von links nach rechts über den Bildschirm. Legt man nun an den y-Eingang ebenfalls ein Signal an, so erhält man das typische Bild eines Oszilloskops (Bild 2).

Natürlich läßt sich der C 64 als Speicheroszilloskop auch für andere Messungen einsetzen. Dies soll ein Beispiel zeigen, bei dem der im C 64 integrierte A-D-Wandler dazu benutzt wird, die Auf- und Entladekurven eines Kondensators zu demonstrieren. Dazu wird der Kondensator über ein Widerstandsnetzwerk mit der Versorgungsspannung (5 Volt), Masse und dem analogen Eingang des Computers verbunden. Zwei Widerstände sorgen nun über einen Schalter dafür, daß sich abhängig von der Schalterstellung der Kondensator auf- beziehungsweise entladen läßt. Werden die beiden Widerstände und der Kondensator entsprechend groß gewählt, ist der Auf- beziehungsweise Entladevorgang so langsam, daß er auf dem Monitor gut verfolgt werden kann.

Alle bisher aufgeführten Beispiele sollten nur andeuten, was mit einem Computer im Physikunterricht alles möglich ist. Das Anwendungsspektrum ist damit aber noch lange nicht erschöpft. Vielmehr existieren zur Zeit schon Hard- und Software-Erweiterungen, mit denen sich speziell der C 64 in allen Bereichen der Physik, Chemie und Elektronik sowie Optik und Wärmetechnik als »Meßdiener« bewährt hat. Der C 64 kann aber nicht nur messen. Genauso gut eignet er sich auch zum »Steuern« und »Regeln« beliebiger Maschinen und Meßeinrichtungen aus den oben genannten Bereichen. So lassen sich zum Beispiel Temperaturen absolut konstant halten oder Gasdrücke messen und regeln. Ein weiteres interessantes Thema im Bereich »Messen«, »Steuern«, »Regeln« ist der Einsatz des Computers zur Robotersteuerung. Von den Möglichkeiten, die daraus resultieren, wird zur Zeit an den Schulen so gut wie kein Gebrauch gemacht. Das wird sich wohl ändern müssen, denn Roboter werden heute in vielen praxisbezogenen Berufssparten eingesetzt, zum Beispiel bei der Auto- oder Stahlindustrie. Aber auch in der chemischen Industrie setzen sich Roboter für Arbeiten, die gesundheitsschädigend sind, durch. In naher Zukunft wird sich ihr Einsatzgebiet noch vergrößern, so daß sich Schulen mit diesem Thema genauso beschäftigen müssen, wie das heute beim Ottomotor oder Diesel-Kreis-Prozeß der Fall ist.

Allen Physiklehrern, die das Thema »Computer« in ihren Unterricht mit einbeziehen wollen, ist das Buch »Computerpraxis bei naturwissenschaftlichen Experimenten« zu empfehlen. Es ist nicht nur für diejenigen gedacht, die den Umgang mit Computern beherrschen, auch jene kommen damit zurecht, die bisher noch nie etwas mit Geräten dieser Art zu tun hatten. Das Buch beginnt mit dem Aufbau und der Funktionsweise eines Mikrocomputersystems, erklärt A-D-Wandler und beschreibt Programme und Selbstbaugeräte für den Unterricht bis hin zum Speicheroszilloskop. Auch biologische, medizinische und chemische Bereiche kommen nicht zu kurz, ja selbst die Herstellung von Platinen wird ausreichend genau erklärt.

Die einzige Anforderung, die an den Lehrer gestellt wird, ist der Wunsch sich mit der »neuen« Technologie Computer auseinanderzusetzen. Ist Interesse von Seiten der Lehrerschaft vorhanden, kann der Computer dabei helfen, auch das Interesse der Schüler zu wecken. Denn das Arbeiten mit dem Computer macht Spaß und fördert dadurch die Aufmerksamkeit und die Mitarbeit eines jeden Schülers.

(ah)

Bezugsquelle: Dr. H. Krönke KG, Postfach 730046, 3000 Hannover

Literatur: Praxis der Naturwissenschaften 34, Heft 1 (Jan.85)

G. H. Göritz und H. J. Scheefer, MNU 38 (1985) 80 bis 87

R. Linke, MNU 38 (1985) 281 bis 287, Gesamtkonzept für die informationstechnische Bildung in der Schule, Schriften des Bayrischen Staatsministeriums, Reihe B, Heft 5

Prof. Dr. R. Lincke (Institut für Experimentalphysik Universität Kiel) Physikalische Experimente mit Mikrocomputern

»Computerpraxis bei naturwissenschaftlichen Experimenten«, Akademie für Lehrerfortbildung, Kardinal-von-Waldburg-Straße 6, 8880 Dillingen, 240 Seiten, 17 Mark

Freie Fahrt für D 595

Endlich ist die echte Simulation einer Verkehrsleitung auf der Modelleisenbahn möglich. Ein Heimcomputer steuert in Verbindung mit einer Schnittstelle von Märklin die Anlage und schon fährt Ihr D-Zug 595 genau nach Plan.

Wie ist es doch so schön in der Presse-Mappe von Märklin geschildert: »Vorbei ist die Zeit, als der mit Elektrotechnik vertraute Vater auf dem Dachboden oder im Keller war, Landschaften gestaltete, Bahnstrecken des Großbetriebes nachempfand, lötete, schraubte und klebte, während der in die Elektronik verliebte Junior im Kinderzimmer vor dem Heimcomputer saß und fasziniert auf den Bildschirm starrte. Märklin-Digital-H0 ist der Weg zurück zum gemeinsamen Spielerlebnis.«

Digitale Spielzeugeisenbahn

Das Herz der Märklin-Anlage ist ein Einplatinencomputer. Seine Aufgabe ist es, den Fahrstrom und die Steuersignale zu mischen. Eine Schaltung (IC) in der Lok trennt diese beiden Teile wieder auf, dabei hat jeder Triebwagen seine eigene Ansprechadresse. Der daraus entstehende Vorteil ist, daß man bis zu 80 Triebwagen mit unterschiedlichen Geschwindigkeiten fahren lassen kann. Bisher konnte man dagegen alle Loks nur mit einer Geschwindigkeit fahren lassen. Auch die Weichen und die Signale lassen sich über die Befehle steuern.

Erst eine Computersteuerung schafft es, daß ein Zug seine Geschwindigkeit reduziert und mit Schrittempo in den nächsten Bahnhof einfährt, wenn er das Vorsignal passiert hat (Bild 1). Dank der konstanten Versorgungsspannung ist die Zugbeleuchtung immer gleich hell. Die Beleuchtung schwankte ja bisher, abhängig von der Zuggeschwindigkeit. Übrigens läßt sich das Licht auch durch einen Befehl ein- oder ausschalten. Auch der Antriebsmotor profitiert vom konstanten Fahrstrom. Durch die Phasenanschnittssteuerung hat der Motor bei geringen Drehzahlen ein größeres Drehmoment. Man kann dadurch einen Zug auch sehr langsam fahren lassen. Der wohl entscheidendste Vorteil ist, daß nur noch zwei Leitungen für den Betrieb der gesamten Anlage notwendig sind. Weil Fahrstrom und Steuercode über dieselbe Leitung fließen, ist die Zeit der dicken Kabelbäume vorbei.

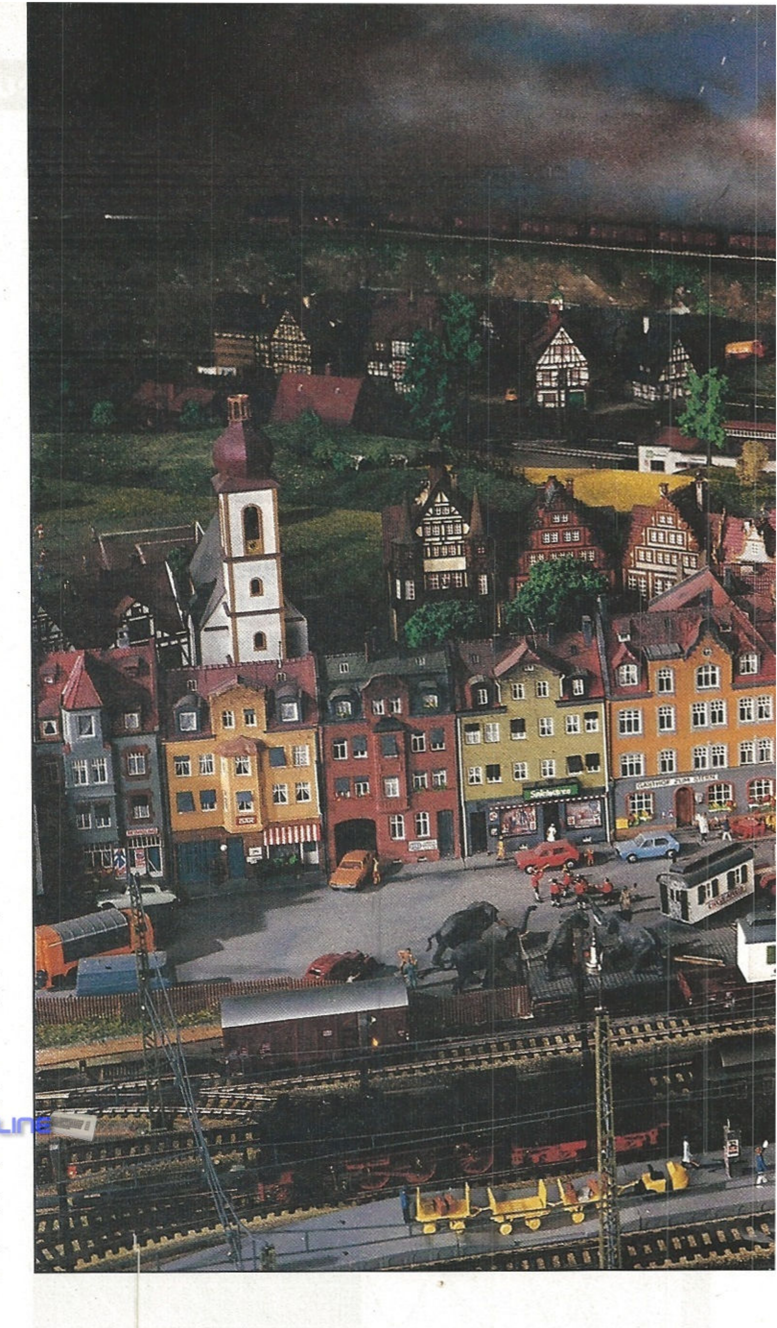
Der Computer steuert

Die Steuereinheit gibt es in zwei Versionen. Zum einen eine manuellgesteuerte und zum anderen eine, die mit einem Computer über eine RS232-Schnittstelle angesteuert werden kann. Die Bausteine:

Der Transformator
Er ist für die Stromversorgung unerläßlich. Der Preis liegt zwischen 70 und 110 Mark.

Der Booster
Ein Leistungsverstärker zum Anschluß von weiteren Transformatoren, die notwendig werden, wenn die Anlage erweitert werden soll (der Verstärker kostet zwischen 70 und 110 Mark).

Das Keyboard
Es ist das Stellpult zum Steuern von 16 Weichen oder Signalen. Für zusätzliche Weichen benötigen Sie ein weiteres Gerät. Sie können bis zu 16 dieser Geräte aneinander koppeln (zwischen 140 bis 170 Mark).



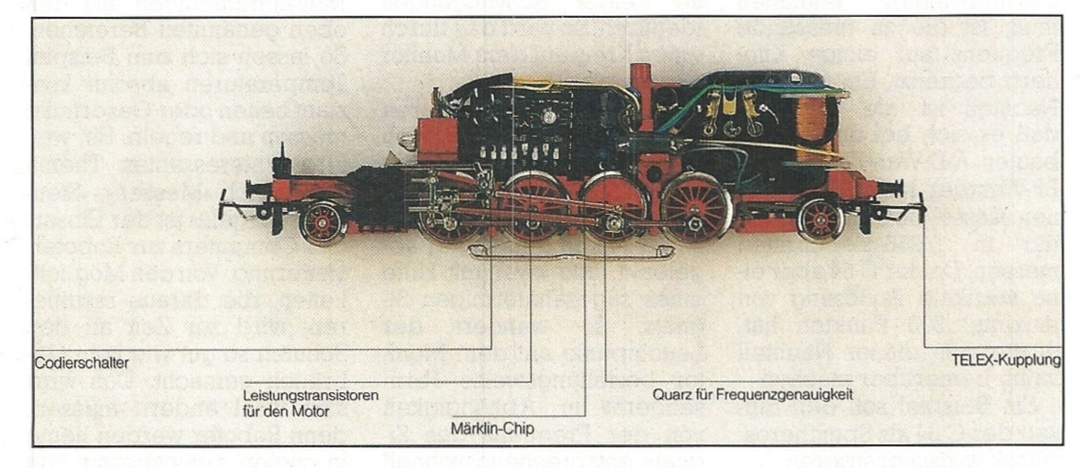


Bild 2. Eine geöffnete Lok mit Modulplatine, Antriebsmotor und Mechanik

Die Zentraleinheit

In der Zentraleinheit befindet sich der Einplatinencomputer, der für die eigentliche Steuerung zuständig ist (zwischen 200 und 300 Mark). Hier wird die Mischung von Fahrstrom und Steuerungssignale vollzogen.

Fahrgerät Control 80

Von hier werden die Triebwagen gesteuert. Mit einer Tastatur kann man alle Triebwagen anwählen und deren Geschwindigkeit regeln. Zu erwähnen ist noch die »Notstop-Taste«, die den Fahrstrom unterbricht, wenn grobe Steuerungsfehler oder ein Zusammenstoß zweier Züge unvermeidbar ist. Ebenso können Sie vom Fahrgerät aus die Sonderfunktionen, wie die Lichtsteuerung im Triebwagen, aktivieren (zwischen 150 und 200 Mark).

Der Decoder

Zur Steuerung der Weichen und Signale benötigen wir den Decoder. Er filtert sich die Signale aus dem Fahrstrom, die er für seine vier überwachten Geräte (Weichen, Signale, Lampen oder ähnlichem) benötigt (zwischen 80 und 110 Mark). Der Bausatz der manuellen Steuerung ist damit komplett und funktionsfähig. Er wird nur noch durch eine Schnittstelle (Interface-Unit) ergänzt, die den Betrieb mit einem Heimcomputer ermöglicht. Die Schnittstelle kostet zwischen 140 und 180 Mark.

Triebwagen mit Gehirn

Die Denkzentrale in jedem Digital-H0-Triebwagen ist der Märklin-Chip 50250 (Bild 2). Er übernimmt die Demodulation, das heißt die Trennung von Fahrstrom und Steuerimpulsen. Die Entwickler der kleinen Platine haben darauf Wert gelegt, daß auch die ältesten Märklinloks auf die Digital-H0-System umrüstbar sind. Es braucht nur der alte Fahrtrichtungsumschalter gegen die neue Platine ausgetauscht werden und schon hat man eine moderne Digital-H0-Bahn. Eine solche Umrüstung kostet zirka 80 Mark.

Über eine RS232-Schnittstelle ist das Interface mit dem Computer (C 64) zu verbinden. Nach dem Aktivieren der Schnittstelle mit OPEN2,2,0,CHR$(10) + CHR$(0) übernimmt der Computer die Steuerung über die Teile der Anlage, die ihm übertragen wurden. Das kann die Steuerung eines Rangierbahnhofes sein. Das Steuerungsmenü muß alle aktuellen Befehle in kurzen Abständen wiederholen, damit ein sicherer Betrieb gewährleistet ist. Man sollte allerdings beachten, daß die Übertragung mit 2400 bit/s bei einem C 64 Übertragungsfehler verursachen kann.

Über den aktivierten Vollduplexbetrieb kann man ständig alle Module, Triebwagen und Schaltelemente abfragen. Sie geben Aufschluß darüber, welcher Zug sich gerade wo auf der Anlage befindet, oder ob ein Befehl richtig ausgeführt wurde.

Das Märklin-H0-System eröffnet eine neue Dimension beim Betrieb einer Modelleisenbahn. Damit ist eine naturgetreue Kopie einer Bundesbahn-Anlage möglich geworden, denn hier ist der Computer schon lange ein Hauptbestandteil im Steuerungssystem. (do)



Bild 1. Musteraufbau einer computergesteuerten Modellbahnanlage mit Fahrpult

Digitalvoltmeter für den C 64/C 128

Wem ein analoges Voltmeter zu ungenau und wer bei Messungen auf genaue Zahlenwerte angewiesen ist, dem bleibt nur die Anschaffung eines Digitalvoltmeters. Erweitern Sie Ihren Computer zum Digitalvoltmeter.

Ein Multimeter ist bei fast jedem Elektronikbastler zu finden. Sie sind recht vielseitig, meist klein und handlich. Sie haben einen großen Drehknopf, zum Einstellen der Meßbereiche und ein Zeigermeßinstrument für die Anzeige. Will man einen Wert, zum Beispiel 2,55 Volt, ablesen, ist man auf das Schätzen angewiesen, denn die Zeigerspitze steht zwischen zwei Strichen. Einen exakten Zahlenwert liefert uns dagegen nur ein Digitalvoltmeter (Bild 1).

Das Herz eines Digitalmeßsystems ist der Analog-/Digital-Wandler. Er übernimmt die Umsetzung einer analogen Spannung in einen digitalen Zahlenwert. Wo liegt der Unterschied zwischen einer digitalen und einer analogen Größe? Ein digitaler Wert kennt nur zwei Zustände, also zum Beispiel 5 Volt oder 0 Volt. Ein analoger dagegen ist nicht fixiert, dieser Wert kann zwischen 0 und unendlich liegen. Um die Funktion eines A/D-Wandlers zu verdeutlichen, greifen wir eine von mehereren Wandlungsarten heraus: Eine Referenzspannung wird in kleinen Schritten erhöht und mit dem Meßwert verglichen. Die Anzahl der Schritte bis zum Ausgleich ergeben den Digitalwert.

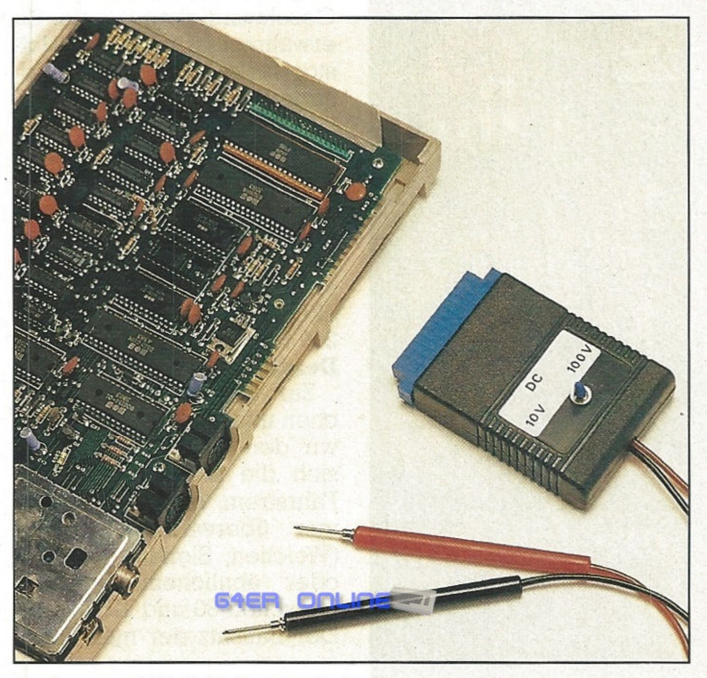


Bild 1. Meßwertaufnehmer mit A-D-Wandler und externen Bausteinen im Gehäuse

Die Meßbereiche sind in der Regel recht klein (zum Beispiel ICL7106 von -200mV bis +200mV), dafür aber recht empfindlich. Die kleinste lesbare Größe liegt bei zirka 100 µVolt. Mit einem Spannungsteiler kann dieser Bereich, auch auf größere Meßwerte erweitert werden. Mit einer Schaltungserweiterung lassen sich Ströme, Wechselspannungen, Widerstände, Kapazitäten und Induktivitäten messen.

Der Computer fragt die Schnittstelle mit dem A-/D-Wandler in regelmäßigen Abständen ab. Er berichtigt den Wert um den Faktor der Spannungsteilung und gibt ihn an den Benutzer aus. Je nach dem, wie oft die Werte ausgelesen werden, können auch Spannungsverläufe sichtbar gemacht werden. Hier liegt aber auch die Grenze der Digitalmeßsysteme, da die Meßzeiten nicht unter 20 Millisekunden gebracht werden können (Taktrate des Computers) oder die Auflösung nicht ausreicht (8 Bit A/D-Wandler). Die Dokumentation der Meßwerte kann über alle Peripherie-Geräte, wie Bildschirm, Drucker, Plotter ausgegeben werden.

(do)

Speicheroszilloskop

Ein Speicheroszilloskop braucht man, um zum Beispiel die Temperaturschwankungen in einem Treibhaus sichtbar zu machen. Wie Sie die Aufgabe von Ihrem Computer ausführen lassen können, wollen wir Ihnen hier zeigen.

Ein Computer kann nur digitale Größen verstehen. Da die Meßwerte meist als analoge Größen vorliegen, müssen wir sie erst in digitale Größen übersetzen. Die notwendige Umsetzung übernehmen Analog/Digital-Wandler (A/D-Wandler). Dabei wird ein analoger Wert in viele kleine Teilstücke zerlegt. Diese »Teilstücke« kann man durch Zahlen darstellen, mit denen der Computer weiter arbeiten kann.

Durch ein Programm können wir den Computer zum Beispiel veranlassen, alle 10 Minuten eine Temperatur zu messen. Diese Werte werden in eine Tabelle der Reihe nach eingetragen und stehen somit für eine spätere Auswertung bereit. Hierbei ist es besonders wichtig zu wissen, wie sich die Temperatur in einem Heizkessel, in einem Aquarium oder Treibhaus ändert, um Geräte, Pflanzen oder Tiere vor Schaden zu bewahren.

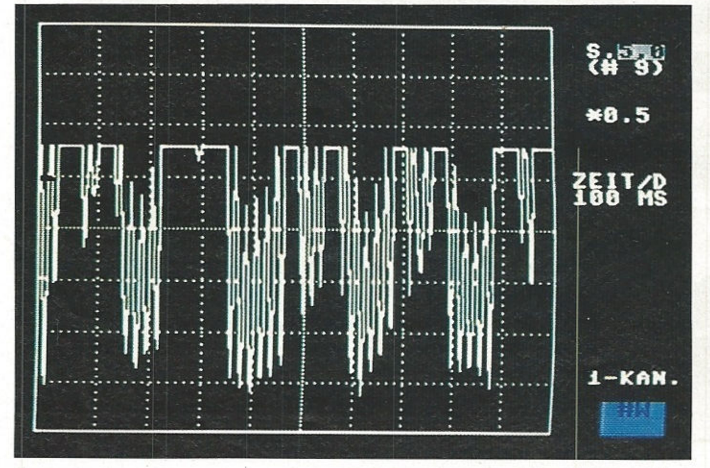


Bild 1. Aufgezeichnetes Tonsignal aus einen defekten Mikrofon

Es ist recht mühsam, sich anhand von vielen Zahlenwerten zeitliche Änderungen eines Wertes vorzustellen. Es bleibt in der Regel nur die Lösung, die Messungen in ein Diagramm einzutragen.

Verbindet man nun die einzelnen Punkte miteinander, so erhält man eine anschauliche Kurve. Diese Arbeit können wir genausogut dem Computer überlassen.

Speichernde Oszilloskope werden heute in allen Bereichen der Geräte- und Maschinenüberwachung eingesetzt. Besonders langfristige Änderungen der Meßwerte entgehen dem Beobachter und es können schwere Beschädigungen an Anlagen entstehen. In der Elektronik kann ein Speicheroszilloskop hilfreich sein, um niederfrequente Tonsignale sichtbar zu machen.

(do)

Die Zeit bleibt auch in der Diskothek nicht stehen

Wie in allen anderen Bereichen unseres Lebens, so hält auch die Computertechnik ihren Einzug in die Disko. Laser-Light-Show ist ein wichtiger Aspekt der jungen Generation bei der Auswahl Ihrer Stammdisko, ebenso wie die präsentierten Hits.

Trotz des riesigen technologischen Fortschrittes in der Computertechnik, sind Diskotheken, die computerunterstützte Licht- und Lasereffekte einsetzen, noch rar gesät (Bild 1).

Um Lasereffekte wie Bilder und grafische Figuren zu erzeugen ist man auf ein kompliziertes Spiegelsystem angewiesen. Die Steuerung dieser Spiegel übernimmt dabei ein Computer.

Über zwei drehbare Spiegel kann der feingebündelte Laser in der X-Achse und Y-Achse bewegt werden, dabei sind beide Spiegel um einen Bereich von 30 Grad schwenkbar (Bild 2). Die jeweilige Position des Spiegels wird über feine Sensoren ermittelt. Ein Steuerungsprogramm verdreht die Spiegel so weit, daß der Laser, ähnlich wie beim Fernseher, eine Fläche zeilenweise aufbaut (Bild 3). Mit dem Unterschied, daß die Helligkeit des Strahls nicht geregelt werden kann. Eine Intensitätssteigerung läßt sich nur durch wiederholtes Abfahren einer Kontur erreichen. Um die verschiedensten Lasereffekte an Wänden und Decken hervorzurufen, muß dem Computer in mühsamer Kleinarbeit gesagt werden, wo er den Strahl hinlenken soll. Die Positionen werden in verschiedenen Registern (X- und Y-Wert) abgelegt und können durch Aufruf einer Codezahl abgerufen werden. Ebenso werden die Bewegungsabläufe durch Punkte in einer Tabelle festgelegt (Bild 4). Der Lightjockey, der für die Bedienung der Lichtanlage zuständig ist, kann abhängig von der Musik die einzelnen Effekte und Grafikbilder über ein Steuerpult abrufen (Bild 5). Natürlich ist es auch möglich bestimmte Rhythmen direkt in Lasereffekte umzuwandeln, wie es von der Lichtorgel her bekannt ist.

(do)

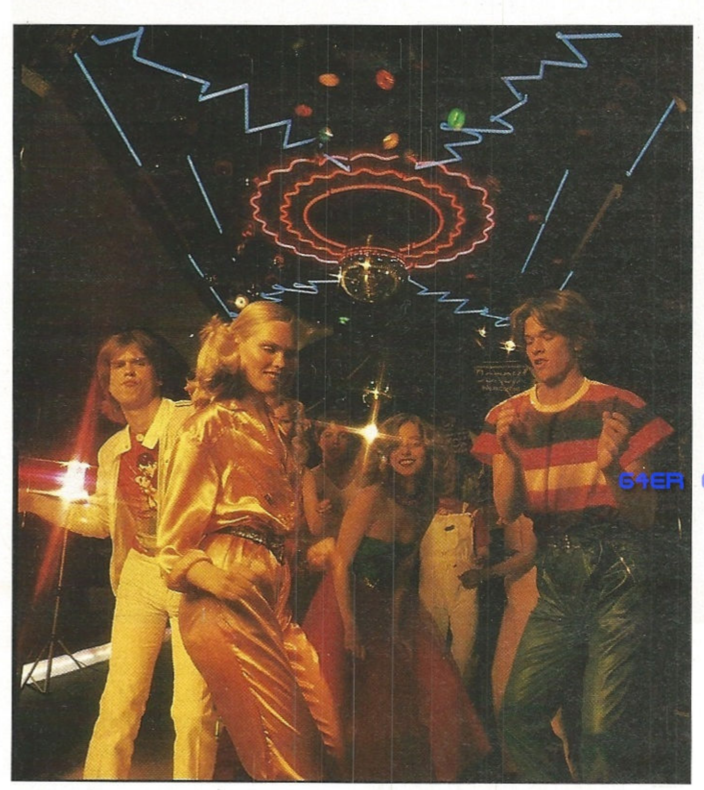


Bild 1. Jugendliche in der Diskothek

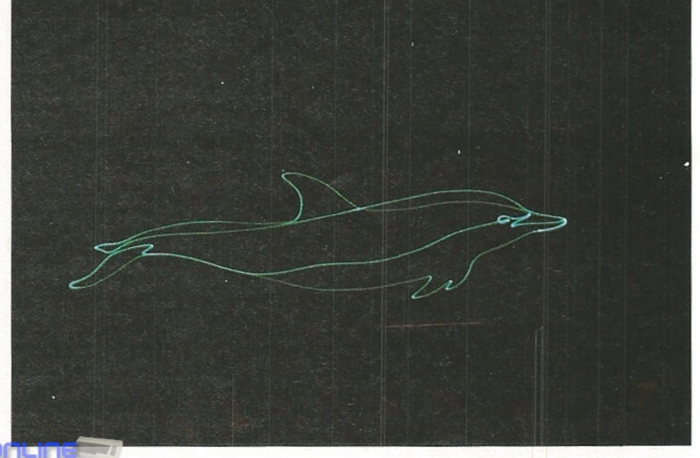
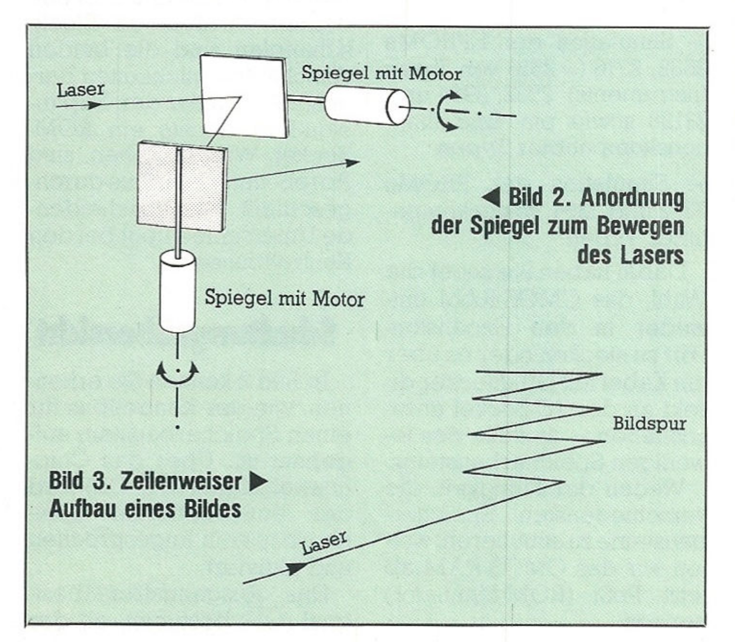


Bild 4. Eine vom Laser gezeichnete Figur (Königstherme bei Augsburg)



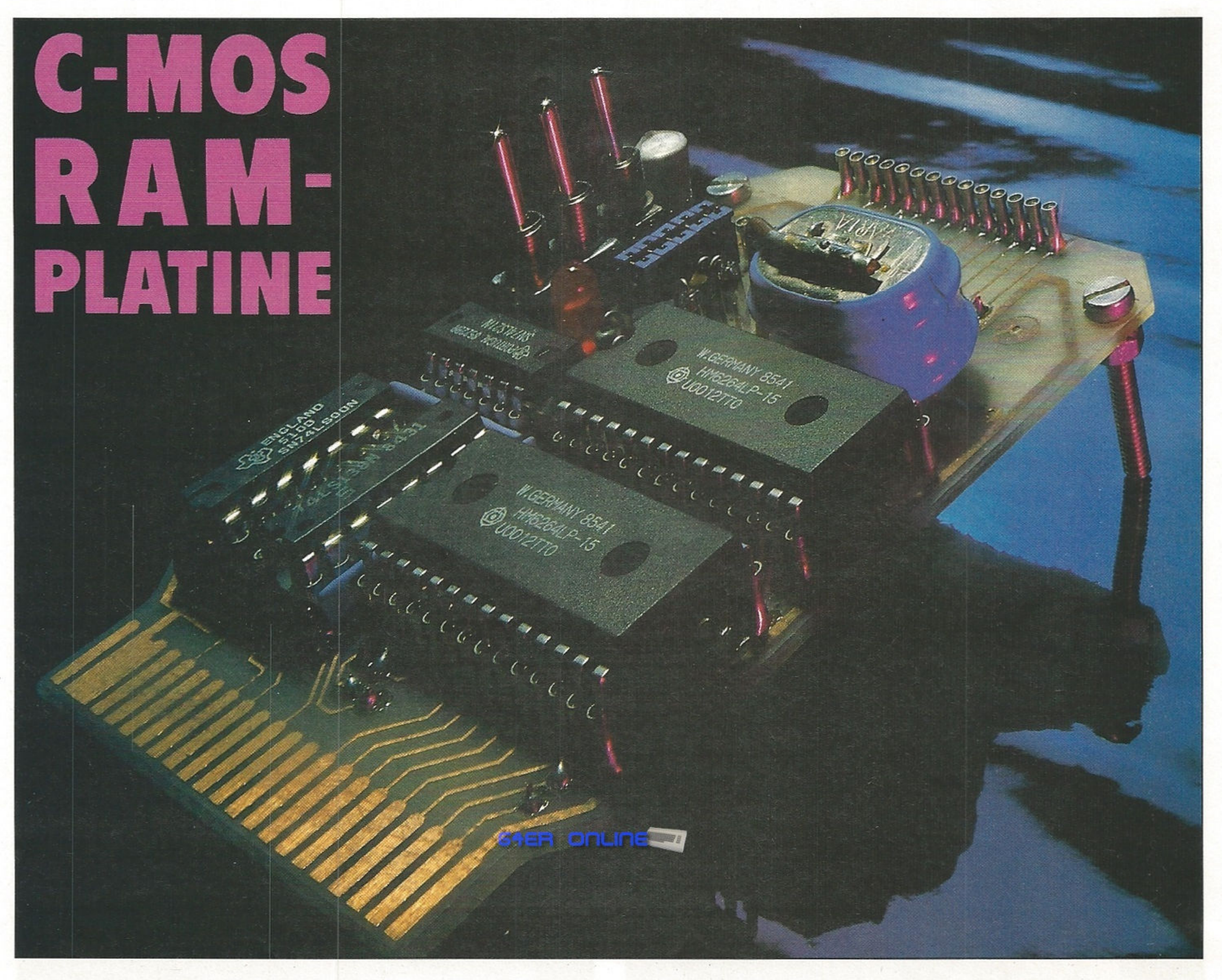
◀ Bild 2. Anordnung der Spiegel zum Bewegen des Lasers

Bild 3. Zeilenweiser Aufbau eines Bildes ▶



Bild 5. Steuerpult einer Laseranlage (Königstherme bei Augsburg)

C-MOS RAM-PLATINE



Haben Sie schon einmal Ihr Programm auf ein EPROM gebrannt und sich anschließend geärgert, weil doch noch eine kleine Verbesserung nötig war? Das CMOS-RAM schafft Abhilfe. Was es damit auf sich hat, wie es funktioniert und wie Sie es selbst bauen können, wollen wir Ihnen erklären.

Wer das Betriebssystem eines Computers schon einmal verändert hat, wird über die beiden Möglichkeiten der Speicherung (Diskette oder EPROM) nicht gerade glücklich gewesen sein. Das EPROM ist für die Entwicklungsphase eine ungünstige Lösung und das Arbeiten mit Diskette erweist sich für diesen Fall als eine aufwendige Angelegenheit. Auch wenn Sie in den nächsten Wochen beispielsweise vorwiegend mit Ihrem Monitor-Programm arbeiten wollen, kann Ihnen das dauernde Laden von Diskette schnell lästig werden. Dafür ein EPROM zu brennen, ist nicht gerade die billigste und schnellste Lösung.

Das akkugepufferte CMOS-RAM (großes Foto) löst die eben genannten Probleme und viele weitere auf überraschend einfache Weise.

Prinzipiell kann das CMOS-RAM wie jedes andere RAM beschrieben und gelesen werden. Es verliert aber, wie ein RAM, beim Ausschalten nicht den Speicherinhalt.

Auf diese Weise können Sie die verschiedensten EPROMs, PROMs und ROMs mit dem CMOS-RAM simulieren. Im einzelnen ergeben sich folgende Möglichkeiten:

— Simulation von ROM-Modulen am C 64 in den Bereichen $8000 bis $9FFF, $A000 bis $BFFF und $8000 bis $BFFF

— Simulation der EPROMs 2532, 2716 (=2516 von Texas Instruments), 2732, 2764 und 27128 sowie pin- und funktionskompatibler Typen

— Simulation der PROMs 2332 und 2364, sowie kompatibler Typen.

Dabei haben Sie sogar die Wahl, das CMOS-RAM entweder in den Expansion-Port zu stecken oder es über ein Kabel mit DIL-Stecker direkt an den IC-Sockel anzuschließen — anstelle des jeweiligen Speicherbausteins.

Wegen der Fähigkeit, die verschiedensten Speicherbausteine zu simulieren, wollen wir das CMOS-RAM ab jetzt RoSi (ROM-Simulator) nennen.

Für den ersten Überblick ist es ratsam, sich an dem Blockschaltplan in Bild 1 zu orientieren. An dieser Stelle wollen wir auch erwähnen, daß alle Abbildungen inklusive Layout mit dem C 64 entwickelt wurden. Im Blockschaltplan sind die beiden Anschlußmöglichkeiten dargestellt — links der Expansion-Port, rechts ein ROM-Sockel. Wie Sie sehen, sind Adreß- und Datenbus durchgeschleift. Der enscheidende Unterschied liegt bei den Kontrollbussen.

Schaltungsübersicht

In Bild 2 können Sie erkennen, wie der Kontrollbus für einen Speicherbaustein aufgebaut ist. Über das Chipauswahlsignal ($\overline{CE}$, CE) wird der entsprechende Speicherbaustein angesprochen und aktiviert.

Das Ausgangsfreigabesignal ($\overline{OE}$) bestimmt, ob der

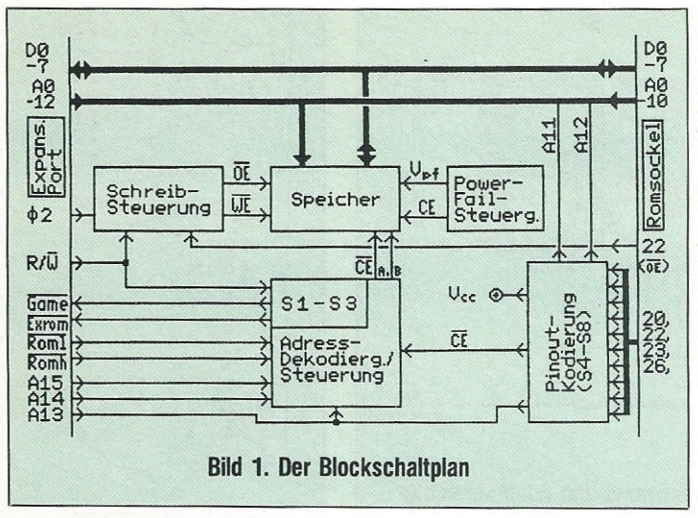


Bild 1. Der Blockschaltplan

Bild 2. Schaltbild des Speicherblocks

adressierte Speicherinhalt auf dem Datenbus erscheint oder nicht.

Bei RAMs kommt gegenüber ROMs noch ein Schreibsignal (WE) hinzu, dessen Logikpegel signalisiert, ob gelesen oder geschrieben wird.

Hier noch eine Anmerkung zu den Signalbezeichnungen. Es wird immer der aktive Logikpegel gekennzeichnet. Ist eine Funktion mit dem Signalpegel L (Low) aktiv, so findet man einen Querstrich über der Signalbezeichnung.

Die Kontrollsignale müssen natürlich vom steuernden System zur Verfügung gestellt werden.

Am Expansion-Port übernimmt der C 64 die Kontrolle. Aus dem Systemtakt Φ_2 und dem R/W-Signal des Prozessors werden im Block »Schreibsteuerung« die Signale OE und WE gewonnen.

Die Adreßleitungen A13 bis A15 werden im Block »Adreß-Decodierung/Steuerung« zu CEa,b decodiert und bestimmen, in welchem Adreßbereich das CMOS-RAM eingeblendet ist. Die Signale GAME, EXROM, ROML und ROMH bilden zusätzliche Bedingungen für die richtige Auswahl des Speicherbausteins.

Ist RoSi direkt mit einem ROM-Sockel verbunden, so wird kein R/W-Signal benötigt. Für die richtige Schreibsteuerung sorgt das OE-Signal an Pin 22.

Mit der Pinout-Codierung wird das RoSi an die unterschiedlichen Pinbelegungen der simulierbaren Speicherbausteine angepaßt.

Schließlich ist da noch der Block »Power-Fail-Steuerung«, der ständig die Betriebsspannung überwacht und daraus zwei Funktionen ableitet. Zum einen bestimmt er darüber, ob der Speicher vom Akku oder vom Computer mit Strom versorgt wird. Bei einer Versorgung durch den Computer wird der Akku gleichzeitig geladen. Zum anderen sorgt der Block dafür, daß beim Unterschreiten der Betriebsspannung (Computer ausgeschaltet) keine Daten abrufbar sind.

Die Schaltung im Detail

Nach dem Überblick nun zu den Details. Fast in allen Teilschaltbildern tauchen einige Widerstände auf, die scheinbar unmotiviert von IC-Logikeingängen nach Plus oder Masse gelegt sind. Dies sind sogenannte »Pull-up«- oder »Pull-down«-Widerstände. Sie sorgen dafür, daß der jeweils angeschlossene IC-Eingang auf einen definierten Pegel gelegt wird. Dabei sind Widerstände mit gleichem Wert zu einem Array (A1, A2 oder A3) zusammengefaßt.

Speicherblock: In Bild 2 sieht man die Schaltung des Speichers. Zwei CMOS-RAMs sind einfach parallel geschaltet. Auf diese Weise ergibt sich eine Speicherkapazität von 16 KByte. Die Chip-Enable-Signale CEa,b müssen allerdings getrennt sein.

Der Pull-down-Widerstand an der Adreßleitung A 12 sorgt dafür, daß bei der Simulation von 4-KByte-ROMs nur der untere Adreßbereich des CMOS-RAMs aktiv ist. Die Pull-up-Widerstände an den Kontrolleingängen sind unbedingt notwendig, um bei ausgeschaltetem Computer einen Verlust der Speicherdaten zu verhindern.

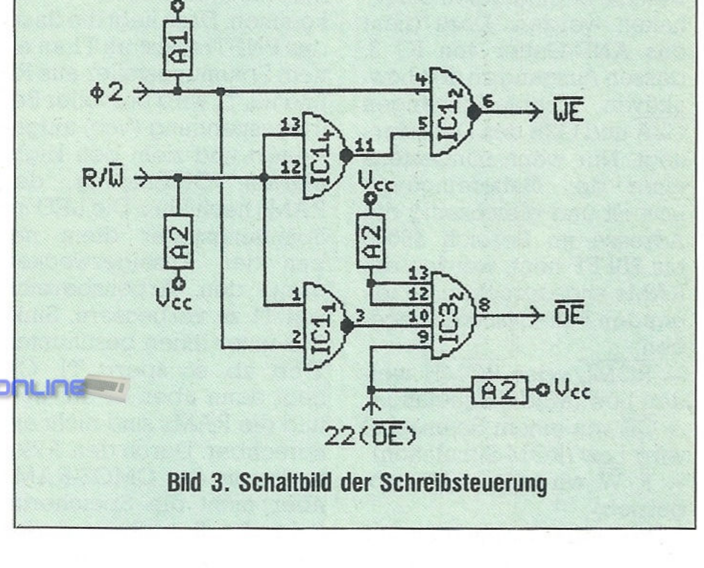


Bild 3. Schaltbild der Schreibsteuerung

Der Stromversorgungsanschluß der ICs erhält hier die Bezeichnung Vpf (normalerweise Vcc), denn so läßt er sich von der Systemversorgung unterscheiden.

Schreibsteuerung: Die Schreibsteuerung (Bild 3) sieht auf den ersten Blick vielleicht etwas konfus aus, läßt sich aber schnell »aufdröseln«. Schreibzugriffe (WE = L) dürfen in einem 6502-System nur auftreten, wenn Φ_2 = H ist. Das wird durch Verknüpfung von Φ_2 und R/W mittels der NAND-Gatter 2 und 4 erreicht. Gatter 4 dient dabei nur als Inverter. Die Pull-up/Pull-down-Widerstände aus den Arrays A2/A1 ziehen WE bei ungesteuerten Gattern auf High. In vielen Systemen werden die OE-Leitungen der Speicherbausteine einfach fest auf Low oder an die entsprechenden CE-Leitungen gelegt. Da die CMOS-RAMs im Schreibbetrieb einfach parallel zu den C-64-RAMs liegen, würde es hier jedoch zwangsläufig zur Datenkollision auf dem Bus kommen. Deshalb wird der Output über das NAND-Gatter 1 nur im Lesebetrieb freigegeben. Das AND-Gatter 2 von IC 3 läßt wahlweise das so gewonnene OE-Signal oder das vom ROM-Sockel kommende OE-Signal zum RAM.

Adreßdecodierung: Nun zur Adreßdecodierung in Bild 4. Die drei höchstwertigen Adreßleitungen A13 bis A15 liegen an den Adreßeingängen eines 3-Bit-Binärdecoders. Dieser macht daraus an seinen Ausgängen (Y0 bis Y7) CE-Signale für jeweils 8-

KByte-Gruppen. Die CMOS-RAMs werden von Y4 und Y5 angesprochen. Dadurch liegen sie im Adreßbereich $8000 bis $BFFF. Es genügt hier nicht, die am Expansion-Port vorhandenen Signale ROML und ROMH zu verwenden, da sie nur im Lesebetrieb wirksam sind. Außerdem ist für die ROM-Simulation ohnehin eine Decodierung notwendig. Die Pull-up/down-Widerstände an den Adreßleitungen sorgen in diesem Fall dafür, daß nur die untere Speicherhälfte angesprochen wird.

Mit der Decodiererei allein ist es aber noch nicht getan, denn es müssen noch weitere Bedingungen eingehalten werden. Dazu dient das AND-Gatter von IC 3, dessen Ausgang an den Low-aktiven Enable-Eingängen G2A und G2B des Decoders liegt. Nur wenn mindestens einer der Gattereingänge Low ist und gleichzeitig die Adresse im Bereich $8000 bis $BFFF liegt, werden die RAMs angesprochen. In folgenden Fällen ist dies gegeben:

— ROML oder ROMH werden Low (Modul-Simulation)
— CE von einem Romsockel wird Low (ROM-Simulation)
— R/W wird Low (Schreibbetrieb)

Die letzte Variante ist nur möglich, wenn der Schreibschalter S3 geschlossen ist.

Die Schalter: Jetzt kommen wir zu den mit EXROM und GAME verbundenen Schaltern S1 und S2 (Bild 5). Sie dienen dazu, den C 64 darüber in Kenntnis zu setzen, ob ein ROM-Modul vorhanden ist und in welchem Adreßbereich es liegt. Die beiden Schalter sind so verdrahtet, daß die verbotene Kombination, EXROM = H und GAME = L, nicht möglich ist.

Die Schaltung des Blocks »Pinout-Codierung« in Bild 7 besteht aus fünf Umschaltern oder Steckbrücken (S4 bis S8) und bestimmt das zu simulierende IC.

Spannungsversorgung: Die »Power-Fail-Steuerung« (Bild 6) ist ebenso einfach wie wirksam. Sie muß bei absinkender Betriebsspannung die CMOS-RAMs schnell genug in den inaktiven Zustand versetzen, damit keine unkontrollierten Informationsänderungen vorkommen. Dazu liegt die Basis des PNP-Transistors T1 an einem Spannungsteiler aus R3 und R2. T1 wird bei voller Betriebsspannung (Vcc) aufgesteuert und zieht den High-aktiven CE-Eingang der RAMs nach Plus. Die LED im Spannungsteiler dient neben den Anzeigezwecken dazu, den Arbeitsbereich von T1 zu verbessern. Sinkt Vcc unter einen bestimmten Wert ab, so sperrt T1. CE liegt dann über R1 auf Low und die RAMs sind nicht ansprechbar. Durch den Akku verlieren die CMOS-RAMs aber nicht die Speicherinhalte. Im Betrieb wird der Akku über D1 und R4 geladen, während er bei fehlender Versorgungsspannung die RAMs über D2 mit etwa 2,2 bis 3,4V versorgt. Mit dem Schalter S9 kann der Akku abgeschaltet werden, um eine Tiefentladung bei längerer Lagerung zu verhindern.

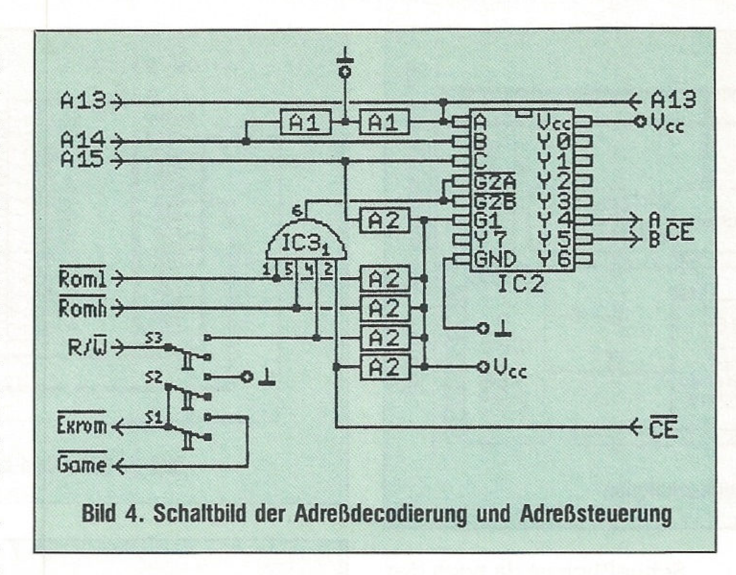


Bild 4. Schaltbild der Adreßdecodierung und Adreßsteuerung

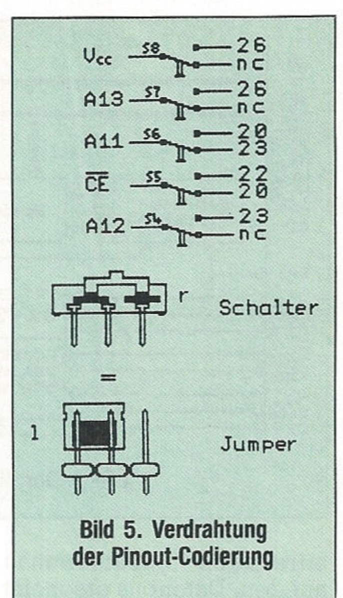


Bild 5. Verdrahtung der Pinout-Codierung

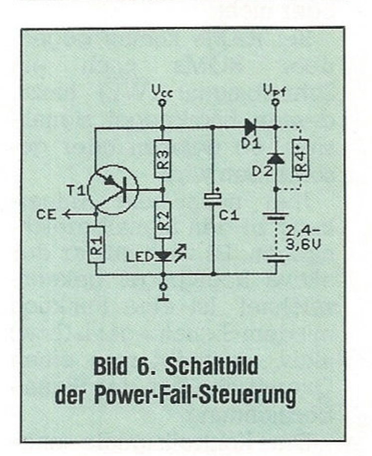


Bild 6. Schaltbild der Power-Fail-Steuerung

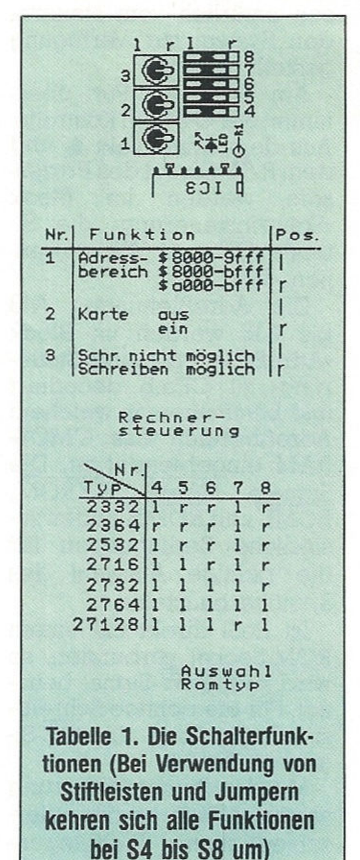
Bauteile

Der Aufbau der Schaltung fängt natürlich mit dem Beschaffen der Bauteile an. Die in Tabelle 4 abgebildete Bestückungsliste gibt Ihnen einen hilfreichen Überblick.

Das Layout für die Herstellung der Platine finden Sie in den Bildern 7 und 8. Es entstand übrigens vollständig auf dem C 64, wie auch alle anderen Bilder.

Die Beschaffung der elektronischen Bauteile (ICs, Widerstände, etc.) dürfte keine größeren Schwierigkeiten machen. Bei der LED muß man sich wegen der erforderlichen Durchlaßspannung an den in der Stückliste angegebenen Typ halten. Für alle ICs empfehlen sich Sockel, nach Möglichkeit Präzisionsausführungen. Bei einigen Bauteilen ist auf ihre Polung gemäß Bestückungsplan (Bild 9) zu achten. ICs haben einen Punkt an Pin 1 oder eine Kerbe an derselben Stirnseite. Die Pinbelegung der anderen gepolten Bauelemente können Sie dem Bild 10 entnehmen. Wie man bei Beschaffungsschwierigkeiten der Widerstandsarrays diese aus acht Einzelwiderständen selbst herstellt, finden Sie ebenfalls dort.

Nun zum Akku. Hier sind die Abmessungen wichtig. Die zur Verfügung stehende Grundfläche auf der Platine beträgt etwa 15x33 mm. Der Pluspol liegt in jedem Fall bei D2, wie im Bestückungsplan ersichtlich, für den Minuspol stehen verschiedene Lötpunkte zur Auswahl. Die Kapazität des Akkus ist nicht so wichtig, da der Stand-By-Strom der Schaltung im Mikroampere-Bereich liegt und damit etwa die Größenordnung der Selbstentladung hat. Ohne weiteres können Sie statt eines Akkus auch eine Trockenbatterie verwenden. Gut geeignet sind hier Lithiumzellen zum Einlöten. Der Betrieb mit Lithiumzellen empfiehlt sich besonders, wenn RoSi nicht häufig benutzt wird. Sie sind auch umweltfreundlich und sichern die Daten über Jahre. Bei Batteriebetrieb kann der Ladewiderstand R4 wegfallen. Für Akkus sollte der Ladestrom etwa 1/50 der



Nr.	Funktion	Pos.
1	Adress-bereich $8000-9fff	l
	$8000-bfff	r
	$a000-bfff	r
2	Karte aus	l
	ein	r
3	Schr. nicht möglich	l
	Schreiben möglich	r

Rechner-steuerung

Typ \ Nr.	4	5	6	7	8
2332	l	r	r	l	r
2364	r	r	r	l	r
2532	l	r	r	l	r
2716	l	l	r	l	r
2732	l	l	l	l	r
2764	l	l	l	l	l
27128	l	l	l	r	l

Auswahl Romtyp

Tabelle 1. Die Schalterfunktionen (Bei Verwendung von Stiftleisten und Jumpern kehren sich alle Funktionen bei S4 bis S8 um)

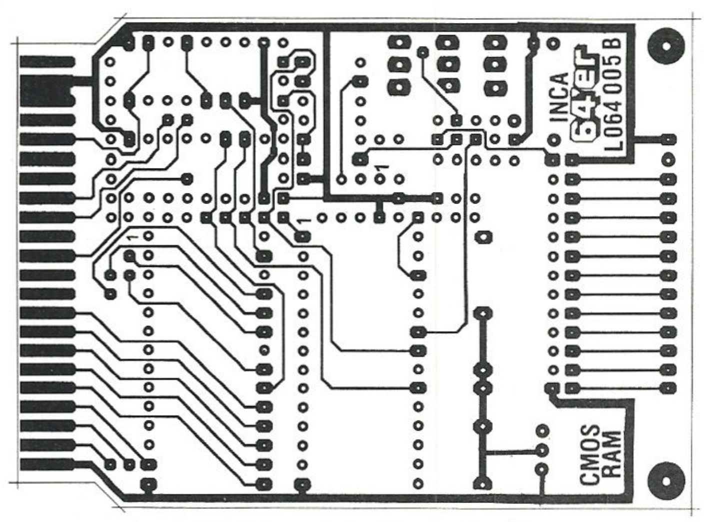


Bild 7. Layout der Platinenoberseite

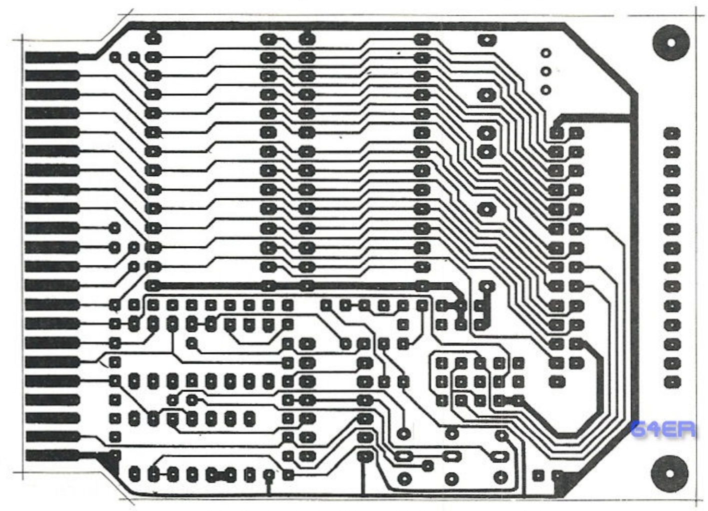


Bild 8. Layout der Platinenunterseite

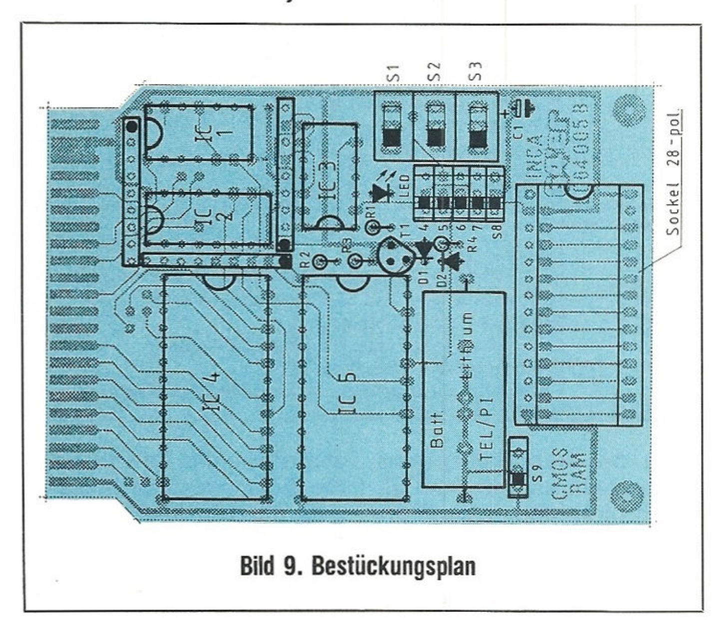


Bild 9. Bestückungsplan

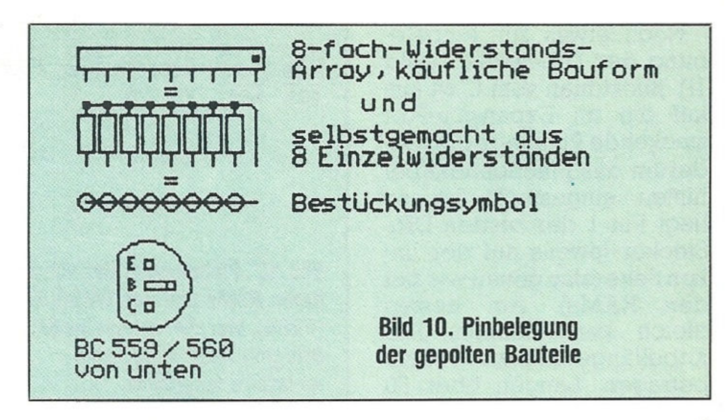


Bild 10. Pinbelegung der gepolten Bauteile

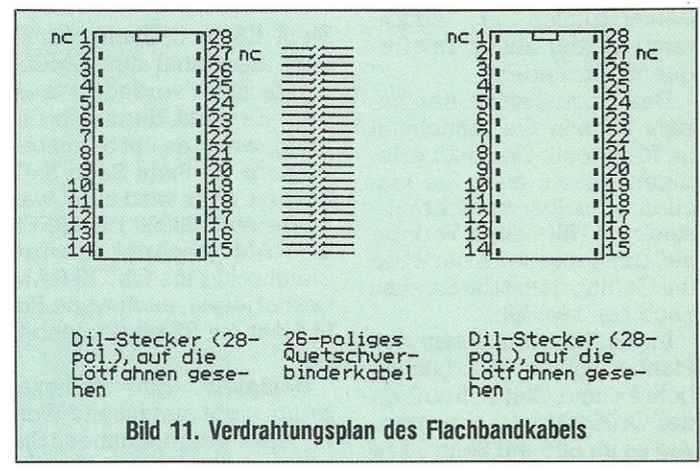


Bild 11. Verdrahtungsplan des Flachbandkabels

Akku-Nennkapazität betragen. R4 errechnet sich demnach wie folgt: R4 (in kΩ) = (4,5V - Ua)/I, wobei Ua die Spannung in Volt des verwendeten Akkus ist und I der gewünschte Ladestrom in mA. Beispiel für Akku 2,4V, 150mAh: I = (150/50)mA = 3mA; R4 = (4,5V - 2,4V)/3mA = 0,7kΩ.

Die Schalter S1 bis S3 sind handelsübliche Schiebe-Umschalter, die anstelle der im großen Foto abgebildeten Kipphebel-Umschalter einzusetzen sind. S4 bis S8 sind einpolige Umschalter im DIL-Format. Diese Spezialbauform ist wegen des gedrängten Aufbaus nötig, jedoch nicht überall erhältlich. Sie können die Schalter durch Stiftleisten mit Steckbrücken (Jumper) ersetzen. Eine Lösung, die preiswerter ist und deren Beschaffung keine Propleme verursacht. Diese Variante ist auch im Foto (Seite 38) zu sehen. Sie müssen aber beachten, daß sich die Schalterfunktionen (siehe Tabelle 1) dabei umkehren.

Es fehlt noch die Steckverbindung über Flachbandkabel zum ROM-Sockel. Probleme dürften bei der Beschaffung des 28poligen IC-Sockels und der DIL-Quetschstecker nebst Flachbandkabel nicht auftreten.

Wenn Ihr Geldbeutel nicht groß genug ist, können Sie auch Sparversionen bauen.

Wenn Sie sämtliche Funktionen benötigen, aber nur EPROMs bis 8 KByte und ROM-Module nur im Bereich \$8000 bis \$9FFF simulieren wollen, können Sie das zweite RAM (IC5) weglassen. In diesem Fall wird auch S1 nicht benötigt. Wer keine Verwendung für die direkte ROM-Simulation hat, kann zusätzlich durch Fortlassen von S4 bis S8 und der kompletten Ausgangssteckverbindung so einiges sparen.

Da keinerlei Abgleich erforderlich ist und sich die Anzahl der Bauteile in Grenzen hält, ist der Aufbau nicht besonders kritisch. Aus dem Bestückungsplan (Bild 9) ersehen Sie die Lage der einzelnen Bauteile. Ein wenig Löterfahrung sollten Sie jedoch schon haben. Wichtig ist auch ein Lötkolben mit dünner Spitze, der wirklich heiß ist. Ein heißer Lötkolben läßt das Zinn schnell fließen und führt zu kurzen Lötzeiten.

Da mit einer zweiten Stromquelle (Akku) gearbeitet wird, sollten Sie besonders auf die richtige Polung der Bauteile achten, um stärkere Rauchentwicklung im Raum zu vermeiden.

Alle Widerstände und Dioden werden stehend eingelötet.

Bei der Bestückungsreihenfolge kann man sich an die Reihenfolge des Abschnitts »Bauteile« halten. In jedem Fall sollte der Akku das zuletzt eingesetzte Bauteil sein!

Ein eventueller Fehler läßt sich mit Hilfe der Schaltungsbeschreibung schnell einkreisen.

Noch etwas zur Pinbelegung der DIL-Stecker (Bild 11). Blickt man vom C 64 aus auf die im Expansion-Port steckende Platine, in die wiederum das Flachbandkabel hinten eingesteckt ist, so liegt Pin 1 der beiden DIL-Stecker jeweils auf der linken Seite (also genau wie bei den RAMs). Am besten gleich kennzeichnen. Die Kabellänge sollte zirka 30 cm betragen. Längen über 50 cm sollten Sie wegen der Kabelkapazitäten in Verbindung mit der hohen Taktfrequenz vermeiden.

Das Anquetschen des Kabels können Sie manchmal im Elektronik-Geschäft erledigen lassen, oder Sie machen es selbst im Schraubstock mit Hilfe einer Vorlage auf der Pinseite. Aber bitte mit Gefühl, damit die Stecker noch heil bleiben.

In die äußeren Ecken der Platine sollten Sie Löcher bohren und Stützschrauben der Größe M3x20 einsetzen, wie es im Bild auf Seite 38 zu sehen ist. Der Expansion-Port-Stecker wird es Ihnen danken.

27128	2764	2732	2716	2532	2364	2332	6264	Pin	Pin (24)	Pin (24)	Pin	6264	2332	2364	2532	2716	2732	2764	27128
Upp	Upp	×	×	×	×	×	NC	1			28	Vcc	×	×	×	×	×	Vcc	Vcc
		×	×	×	×	×	A12	2			27	WE	×	×	×	×	×	PGM	PGM
							A 7	3	1	24	26	CS2	Vcc	Vcc	Vcc	Vcc	Vcc	NC	A13
							A 6	4	2	23	25	A 8							
							A 5	5	3	22	24	A 9							
							A 4	6	4	21	23	A11	CS2	A12	Upp	Upp	A11	A11	A11
							A 3	7	5	20	22	OE	CS1	CS	CS	OE	OE	OE	OE
							A 2	8	6	19	21	A10							
							A 1	9	7	18	20	CS1	A11	A11	A11	CS	CS	CS	CS
							A 0	10	8	17	19	D 7							
							D 0	11	9	16	18	D 6							
							D 1	12	10	15	17	D 5							
							D 2	13	11	14	16	D 4							
							GND	14	12	13	15	D 3							

Bild 12. Pinbelegung der simulierbaren ROMs und des CMOS-RAMs (6264). Nur die unterschiedlich belegten Anschlüsse sind mehrfach aufgeführt.

So gehen Sie mit RoSi um

Nun wollen wir erläutern, wie Sie mit RoSi arbeiten können. Fangen wir mit der Modul-Simulation an.

Modul-Simulation: Für diese Betriebsart wird die Platine in den Expansion-Port des ausgeschalteten C 64 gesteckt, mit der Bauteileseite nach oben. Schalten Sie den Computer jetzt wieder ein. Die Schalter S1 und S2 bestimmen nun die logischen Pegel an den Leitungen EXROM und GAME des C 64. Wie schon erwähnt, sind sie so verschaltet, daß die Kombination EXROM = 1 und GAME = 0 nicht einstellbar ist, denn dies würde unweigerlich zu einem Absturz beim C 64 führen. Nun zu den Funktionen der Schalter. S2 schaltet RoSi aus (l) oder ein (r). S1 bestimmt den Adreßbereich, in den die Karte eingeblendet wird (siehe Tabellen 1 bis 3).

Schalter S3 entscheidet unabhängig von S1 und S2, ob ein Schreibzugriff im Adreßbereich $8000 bis $BFFF auf das CMOS-RAM möglich ist. Steht S3 in Stellung »l«, so kann der Inhal des CMOS-RAMs nicht verändert werden. Es wirkt dann wie ein ROM, wenn es mit S2 eingeblendet ist. Steht S3 in Stellung »r«, dann wird alles, was im Bereich $8000 bis $BFFF ins RAM geschrieben wird, gleichzeitig ins CMOS-RAM geschrieben, auch wenn RoSi nicht mit S2 eingeblendet ist.

Beispiel: Ein Beispiel bringt mehr als tausend Worte. Also: Angenommen, Sie wollen ein 16 KByte langes Autostart-Programm (beispielsweise eine in Maschinensprache geschriebene Textverarbeitung) in RoSi unterbringen, um es jederzeit bei Bedarf einblenden zu können. Dazu stellen Sie S2 auf »l« (RoSi ausgeblendet und S3 auf »r« (Schreiben möglich). Jetzt greifen Sie zu Ihrer Programmdiskette, laden das Programm mit `LOAD "Name",8,1` (die Startadresse $8000 muß natürlich stimmen). Danach stellen Sie S3 wieder auf »l« (Schreiben nicht möglich), um RoSis Inhalt zu sichern. Nun haben Sie ein Modul, das sich exakt wie eine der handelsüblichen EPROM-Bänke verhält. Blenden Sie den Bereich $8000 bis $BFFF mit S1 und S2 ein und — Ihr C 64 stürzt ordnungsgemäß ab, denn der Prozessor »fällt« aus dem Basic irgendwohin in das Programm. Wenn das Autostartprogramm in RoSi in Ordnung ist, und Sie glücklicher Besitzer eines Resettasters sind, bringt nun ein Reset das Programm zum Laufen.

Hatte Ihr Programm keinen Autostart, so können Sie mit dem SYS-Befehl das Programm starten. Sie dürfen dabei allerdings nicht den Bereich $A000 bis $BFFF eingeblendet haben, weil dann der Basic-Interpreter abgeschaltet ist und der SYS-Befehl nicht mehr ausgeführt werden kann.

Exrom	1	1	0	0
Game	1	0	1	0
S1	x	nc	l	r
S2	l	nc	r	r

x =don't care
nc =not connected

Tabelle 2. Zusammenhang zwischen EXROM/GAME und den Schalterstellungen von S1, S2

	$8000-$9fff	$a000-$bfff	$8000-$bfff
S1	l	r	r
S2	r	r	r
Hiram	1	1	1
Loram	1	0	1

Tabelle 3. Zusammenhang zwischen HRAM, LRAM, S1, S2 und den eingeblendeten Adreßbereichen

Bestückungsliste

Halbleiter	
IC 1	74 HC 00
IC 2	74 HC 138
IC 3	74 LS 21
IC 4, 5	6264
T 1	BC559 oder BC560
D 1, 2	AA119 oder ähnlich
LED	5 mm Leuchtdiode, rot
passive Bauteile	
C 1	Elko stehend 47µF/10V
A 1	Widerstandsarray 8 x 4,7 kΩ
A 2, 3	Widerstandsarray 8 x 47 kΩ
R 1	Widerstand 270 kΩ
R 2	Widerstand 270 Ω
R 3	Widerstand 68 Ω
R 4	Widerstand berechnen, je nach Akku
sonstige Bauteile	
S 1, 2, 3	Schiebeschalter 1 x Um
S 4–8	5 x DIL-Umschalter oder 15pol. Stiftleiste mit 5 x Jumper
Akku	Einlötakku 2, 4–3, 6 V oder entsprechende Trockenbatterie (Lithiumzelle)
ST	2 x 28polige DIL-Quetschstecker und 28polige Quetschverbinderkabel ca. 30 cm 2 x 14polig, 1 x 16polig und 3 x 28poliger IC-Sockel

Tabelle 4. Liste der notwendigen Bauteile

Fast genauso einfach ist es, RoSis Inhalt auf Diskette zu speichern. Allerdings empfiehlt sich hierfür ein Monitor, da man in Basic nicht beliebige Adreßbereiche speichern kann. Für das obige Beispiel ergibt sich folgende Vorgehensweise:

Laden Sie einen Monitor, der außerhalb des Bereichs $8000 bis $BFFF liegt und starten Sie ihn. Jetzt blenden Sie RoSi ein, indem Sie S1 und S2 auf »l« legen. Nun speichern Sie den Inhalt von RoSi mit `s"name",08,8000,c000` ab. Je nach Monitor kann auch eine andere Syntax notwendig sein.

Eine prinzipiell gleiche Vorgehensweise ergibt sich für Teile dieses Adreßbereichs, nur die Stellung der Schalter S1 und S2 und der Prozessor-Port müssen jeweils angepaßt werden (Tabellen 1,2 und 3). Wenn man

zum Beispiel nur im Bereich $8000 bis $9FFF operiert, braucht man nicht mal unbedingt einen Monitor, weil das Basic eingeschaltet bleibt. Ein paar Experimente beseitigen schnell alle Unklarheit, so daß Ihrer selbstgemachten Basic-Erweiterung nichts mehr im Wege steht.

ROM-Simulation

Die Pinbelegung der ROMs und EPROMs, die mit RoSi simuliert werden können, finden Sie in Bild 12 neben der des CMOS-RAMs aufgeführt. Es sind nur die Anschlüsse mehrfach abgebildet, die bei den verschiedenen Typen unterschiedlich belegt sind.

Die Handhabung der ROM-Simulation ist ebenso einfach wie die der Modul-Simulation. Zuerst wird der Inhalt des zu simulierenden ROMs nach der Anleitung im vorigen Abschnitt in RoSi untergebracht. Die Stellung der Schalter S4 bis S8 ist hier noch unerheblich. Jede der in Tabelle 1 aufgeführten Kombinationen ist möglich. Ein Tip: die Platine wurde so ausgelegt, daß das am meisten verbreitete EPROM 2764 eine leicht zu merkende Einstellung hat, nämlich S4 bis S8 alle in Stellung »1«. Wenn die Software in RoSi untergebracht ist, wird sie wieder mittels S3 geschützt. Anschließend müssen Sie den Computer ausschalten und die Platine aus dem Expansion-Port ziehen.

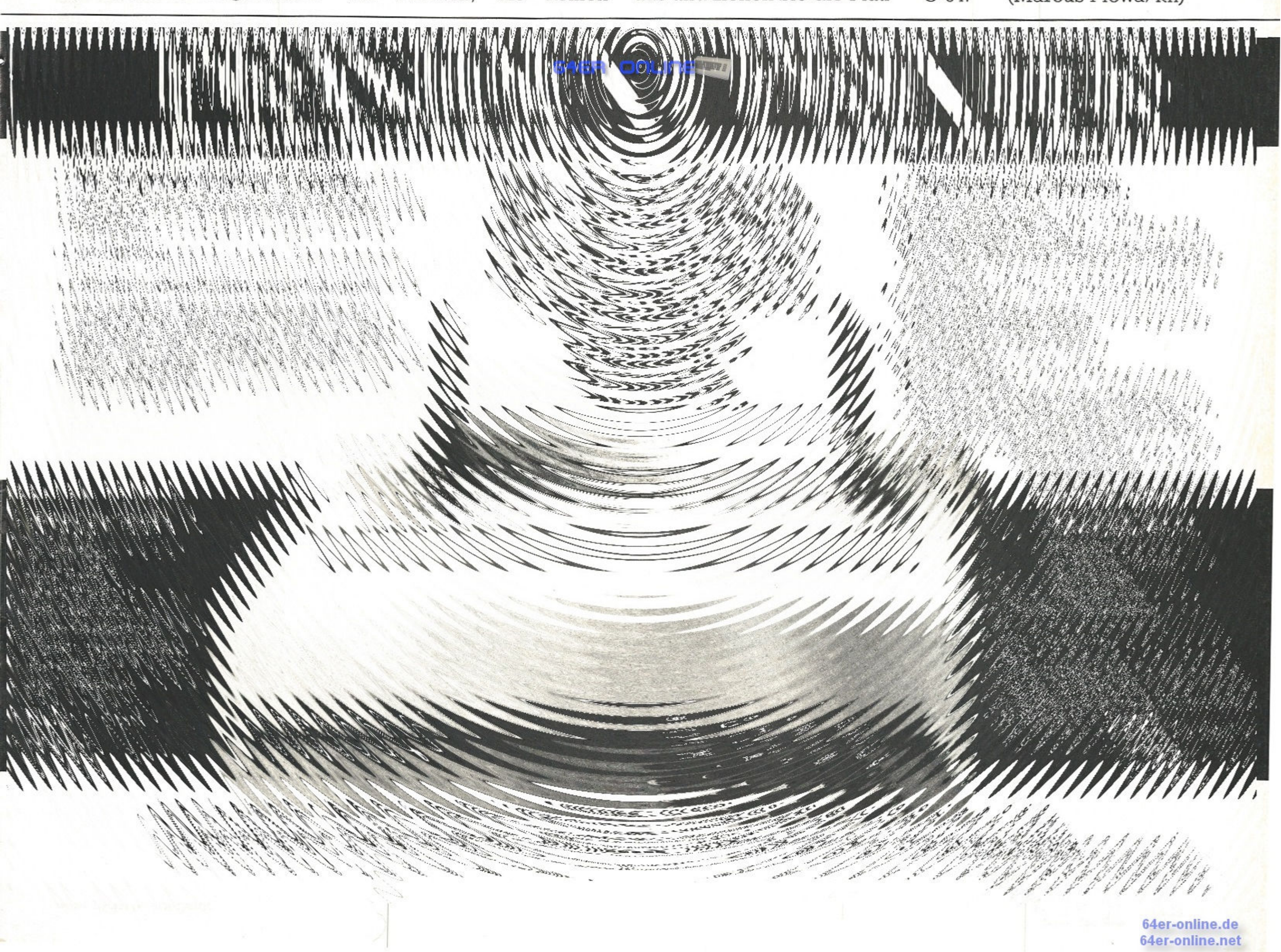
Nie darf RoSi gleichzeitig mit dem Expansion-Port und über Kabel mit einem ROM-Sockel verbunden sein!

Jetzt stellen Sie nach Tabelle 1 den gewünschten ROM-Typ mit S4 bis S8 ein. Das Flachbandkabel stecken Sie bei ausgeschaltetem Gerät mit dem DIL-Stecker in den Sockel des zu simulierenden ROMs. Handelt es sich um einen 24poligen Sockel, so wird der Stecker bündig mit der dem Pin 1 gegenüberliegenden Seite eingesteckt, so daß die Pins 1,2,27 und 28 überstehen. Bei engbestückten Platinen, die keinen überstehenden Stecker erlauben, können Sie sich mit einem 24poligen IC-Sockel behelfen, der noch zwischen den leeren Sockel und den DIL-Stecker eingesteckt wird. Jetzt stellen Sie die Verbindung mit RoSi her und schalten das Gerät wieder ein. Das war's!

Auch hierfür wieder ein Beispiel.

Beispiel: Angenommen, Sie wollen das Schriftbild Ihres Druckers ändern. Sie schalten ihn aus und zerlegen ihn, bis die Platine vor Ihnen liegt. Auf dieser befindet sich, wie das Leben so spielt, ein ROM-Sockel mit einem EPROM des Typs 2764. Hocherfreut hebeln Sie dieses aus der Fassung und lesen seinen Inhalt mit einem EPROMmer oder mit Hilfe von Monitor und EPROM-Bank in den C 64 ein. Mit dem Monitor können Sie nun den Zeichensatz suchen und modifizieren. Dann wird er in RoSi untergebracht. Schalten Sie jetzt den Computer aus und ziehen Sie die Platine aus dem Expansion-Port. Nachdem Sie den EPROM-Typ 2764 eingestellt haben, stellen Sie die Verbindung zwischen RoSi und dem ROM-Sockel im Drucker her. Dann schalten Sie den Drucker wieder ein und — wenn Sie nicht versehentlich das Betriebssystem mit geändert haben — können Sie nun bei einem Probedruck den neuen Zeichensatz bewundern. Sagt er Ihnen nicht zu, so läßt sich der Zeichensatz bis zu Ihrer Zufriedenheit mit Hilfe von RoSi weiterbearbeiten. Abschließend wird er in ein EPROM gebrannt und eingebaut.

Auf gleiche Weise können Sie natürlich auch im C 64 ROM-Simulationen vornehmen. Eine kleine Einschränkung gilt für den PROM-Typ 2332. Um den Schaltungsaufwand nicht unnötig zu erhöhen, wurde vorausgesetzt, daß der CE-Anschluß dieses PROMs in der Schaltung fest auf High liegt, was normalerweise gegeben ist — auch im C 64. (Marcus Plewa/kn)



Platinenätzen

Wir zeigen Ihnen, wie man Platinen leicht selbst herstellen kann. Wie kommt eine Kopie der Leiterbahnen aus einer Zeitschrift auf das Kupfer der Platine? Anschließend finden Sie einige Tips zum Löten. Selbst wenn Sie noch nie einen Lötkolben in der Hand hatten: Es ist einfacher als Sie denken.

Viele Leser fragten uns, wie sie denn die Platinenlayout-Vorschläge aus dem 64'er auf eine Platine übertragen können. Wir wollen Ihnen deshalb an einem einfachen Beispiel einmal zeigen, wie eine Platine geätzt wird. Als Beispiel soll das RS232-Interface aus Ausgabe 3/85 dienen (Bild 1). Denn die Herstellung einer Platine zu dieser Schaltung kann auch vom Anfänger in Sachen Platinenätzen bewältigt werden.

Bevor Sie nun ans Werk gehen, einiges zur Theorie. Eine Platine besteht aus einem Trägermaterial, das der ganzen Platte die mechanische Festigkeit gibt. Als Trägermaterial findet Pertinax (Hartpapier) oder Epoxid-Harz Verwendung. Beide Materialien sind sehr schlechte elektrische Leiter. Auf dem Trägermaterial ist eine Schicht aus Kupfer aufgebracht, die so geätzt wird, daß nur noch die Leiterbahnen zur Verbindung der Bauteile untereinander übrig bleiben. Bei umfangreichen Schaltungen verwendet man »doppelt-kaschierte« Platinen. Hier ist auf der Ober- und der Unterseite der Platine eine Kupferschicht. Man kann damit prinzipiell zwei Platinen auf dem gleichen Träger herstellen. Die Herstellung von doppelseitigen Platinen ist allerdings eine Sache für ausgebuffte Profis.

Epoxidharz hat als Trägermaterial bessere Eigenschaften als Pertinax. Es ist verwindungssteifer und läßt sich exakter bohren. Leider ist es deutlich teuerer als Pertinax.

Bei einer Variante der käuflichen Platinen ist auf das Kupfer noch eine Schicht aus lichtempfindlichen Lack aufgetragen, der das Kupfer vor dem Ätzmittel schützt. Die Lackschicht wird durch eine dicke schwarze Adhäsionsfolie vor Lichteinwirkung geschützt. Setzt man den Photolack UV-Licht (Höhensonne) aus, verändert er seine Struktur so, daß er sich in Natronlauge auflöst.

Für ganz Neugierige erst ein kurzer Überblick über den Werdegang einer Platine: Durch Belichten und anschließendes Entwickeln der Platine bleiben die späteren Leiterbahnen vom Photolack bedeckt. Nach dem Entwickeln wird das freie Kupfer, das nicht mehr vom Lack geschützt wird, weggeätzt. Lediglich die unbelichteten Kupferstellen bleiben stehen. Danach müssen nur noch die Löcher für die Bauteile gebohrt und die Bauteile verlötet werden. Für die Herstellung einer Platine sind ein paar Dinge unverzichtbar (Bild 2). Man kann sie in jedem Elektronikgeschäft bekommen. Im einzelnen:

1. Einseitig beschichtete photopositiv oder negativ (je nach Verfahren) beschichtete Platine
2. Folie zum Übertragen des Layouts (Klebetechnik oder Color Key)
3. Eine Metallsäge und mittelgrobe Feile
4. Ein Tütchen Entwickler (meist Ätznatron)
5. Eine Tüte Ätzmittel (am besten Eisen-III-chlorid)
6. Zwei Plastikschalen (größer als Platine!)
7. Eine dünne Glasscheibe (größer als Platine!)
8. Eine UV-Lampe (notfalls auch Nitraphot 500W)
9. Eine Plastikklammer
10. Bohrer 1 mm und Bohrmaschine
11. Lötkolben (maximal 30 Watt)
12. Einen kleinen Seitenschneider
13. Elektroniklot
14. Natürlich die Bauteile

Um ein Platinenlayout aus dem 64'er-Magazin auf die Platine zu übertragen, gibt es mehrere Verfahren.

Das erste, das Profis sicherlich bekannt ist, heißt »Color Key«. Hier wird eine spezielle Folie auf das Platinenlayout gelegt und mit UV-Licht belichtet. Der Witz dabei ist, daß die Folie nur auf das Licht anspricht, das vom Papier reflektiert wird und nicht auf das Licht der Lampe. Nach dem Entwickeln hat

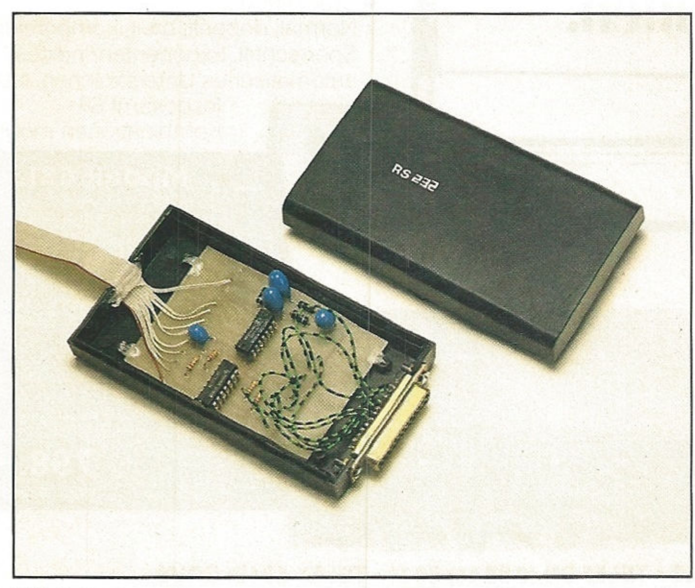


Bild 1. So kann das fertige Interface in einem Gehäuse aussehen



Bild 2. Das sind die benötigten Materialien und Bauteile

leichtgemacht

man eine negative Kopie des Layouts auf Klarsichtfilm, die sich leicht auf eine photonegativ (!) beschichtete Platine übertragen läßt. Hat man photopositiv beschichtete Platinen, muß der Film umkopiert werden. Das heißt, das ganze Spielchen noch mal von vorne. Nur diesmal mit der negativen Kopie als Vorlage. Der Nachteil dieses Verfahrens soll nicht verschwiegen werden: Eine etwa DIN-A4-große Folie kostet acht Mark und 100 ml Entwickler etwa noch mal das gleiche. Hat man noch keine Routine im Color Key-Verfahren, sollte man sich erst mit kleinen Probestreifen, die nicht auf Platine übertragen werden, einarbeiten. Der Vorteil von Color Key ist, daß die Rückseite des Originalplatinenlayouts ruhig bedruckt sein kann.

Color Key und Klebetechnik

Der Anfänger, der sich sowieso nur auf einfache Platinenlayouts beschränken wird, kann das Platinenlayout natürlich auch abzeichnen. Mit normaler Tusche und Zeichenfolie ist das allerdings ein fast aussichtsloses Vorhaben. Erstens verschmiert die Tusche leicht und zweitens ist die Tusche erst nach dem zweiten oder gar dritten Auftrag genügend lichtdicht. Es gibt aber Selbstklebesymbole (Letraset-Verfahren, Abreibe-Buchstaben) die leicht auf eine feste Klarsichtfolie übertragen (abgerubbelt) werden können. Mit diesen Symbolen hat man schnell die Lötaugen für IC-Sockel etc. auf der Folie. Man braucht ja nur die Folie auf die Vorlage legen und diese »nachkleben«. Sehr gut eignen sich Folien für Overhead-Projektoren, die es in den meisten Schreibwarengeschäften gibt. Für Leiterbahnen gibt es Selbstklebebänder auf Rollen. Mit einem kleinen Universalmesser kann das Band leicht auf die passende Länge geschnitten werden. Die Bänder können auch in Kurven verklebt werden. Einfache Layouts sind mit dieser Methode schnell, leicht und sicher herzustellen.

Haben Sie eine Kopie des Platinenlayouts auf Klarsichtfolie (Bild 3), können Sie Ihr handwerkliches Geschick unter Beweis stellen. Das gekaufte Stück Platine muß nämlich auf die Größe der Vorlage gebracht werden. Dazu nehmen Sie eine scharfe Metallsäge. Beim Sägen sollte die Kupferseite der Platine oben liegen. So vermeidet man ein allzu starkes Ausreißen an der Schnittkante. Die Sägekanten lassen sich mit einer Feile schnell entgraten. Achtung! Die Lackschicht darf keine Kratzer bekommen. Denn ungewünschte Leiterbahnunterbrechungen sind die Folge.

Jetzt haben Sie den lästigen Teil der Platinenherstellung hinter sich. Bei dem, was jetzt kommt, ist es besonders wichtig, alles bei der Hand zu haben. Fangen wir mit dem Aufbau für die Belichtung an. Sie benötigen eine ebene Unterlage, über der in 30 cm Abstand die Lampe hängt. Entfernen Sie jetzt die schwarze Schutzfolie vom abgesägten Stück Platine (Bild 4) und legen Sie die Klarsichtfolie seitenrichtig auf die beschichtete Kupferseite (Schrift auf dem Layout muß lesbar sein). Mit einer sauberen Glasscheibe wird die Folie auf die Platine gepreßt. Nun können Sie die Platine mit der darüberhängenden Lampe belichten (Bild 5). Die Belichtungszeit hängt stark davon ab, wie groß der Abstand Lampe — Platine ist und aus welcher Glassorte die Glasplatte besteht etc. In der Regel liegt die Belichtungszeit aber bei etwa 12 bis 15 Minuten. Viel schneller geht es, wenn statt einer Nitraphotlampe ein richtiger UV-Brenner (Höhensonne) verwendet wird. Da die UV-Brenner aber schwer zu beschaffen und teuer sind, möchten wir auf diese Art der Belichtung nicht eingehen. Am besten, Sie sägen kleine Platinenstücke ab und machen ein paar Probebelichtungen, die Sie entwickeln und ätzen. Die einmal ermittelte Zeit paßt dann für alle folgenden Belichtungen, solange Sie die gleiche Lampe im gleichen Abstand zur Glasplatte und Vorlage nehmen. Besorgen Sie sich auf jeden Fall eine passende Fassung für die verwendete Birne. Denn eine 500-W-Birne darf keinesfalls in einer Schreibtischleuchte betrieben werden!

Während die Belichtung läuft, können Sie den Entwickler ansetzen. Lösen Sie

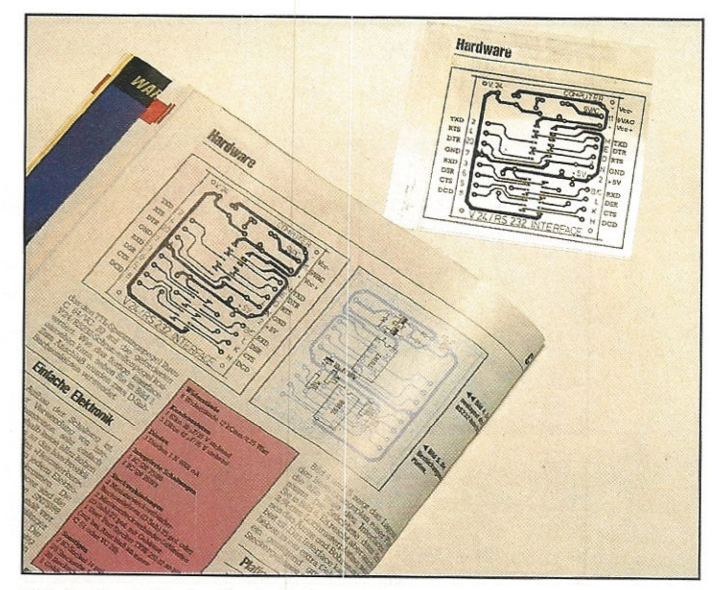


Bild 3. So sieht das kopierte Layout aus

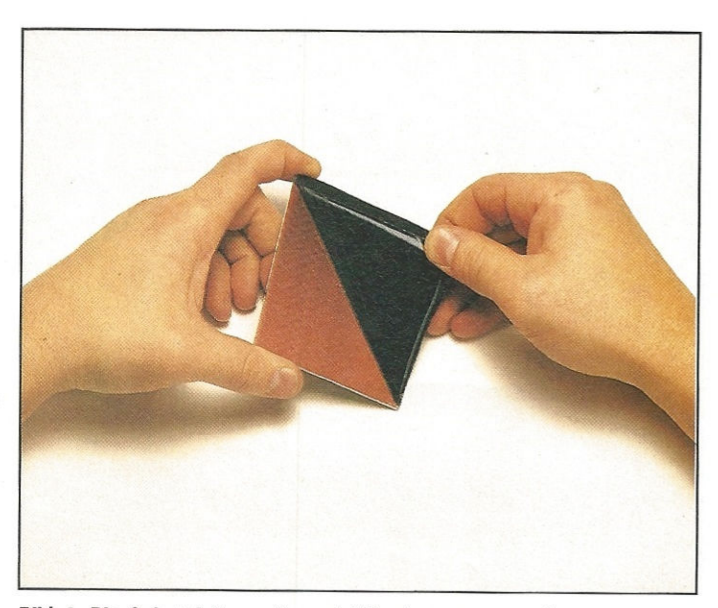


Bild 4. Die Schutzfolie muß vorsichtig abgezogen werden

dazu den Inhalt des Entwicklerpäckchens in der angegebenen Menge Wasser (meist 1 Liter) auf. Die Lösung können Sie aufbewahren und für mehrere Platinen verwenden. Nehmen Sie zur Aufbewahrung aber keine Limo- oder andere Getränkeflaschen, denn der Entwickler ist giftig und ätzend.

Ist die Belichtungszeit vorüber (Stoppuhr), entfernen Sie die Glasplatte und die Klarsichtfolie. Sie können schon ganz schwach die Leiterbahnen auf der Platine erkennen. Fassen Sie nun die Platine an einer Ecke mit einer Wäscheklammer an und bringen Sie die Platine, mit der Schichtseite nach oben, in den Entwickler. Dann sollten Sie die Platine wieder loslassen. Während des Entwickelns und Ätzens sollte die Arbeitsunterlage mit einer dicken Lage Zeitungspapier abgedeckt werden. Selbst zieht man sicherheitshalber einen alten Kittel oder Schürze an. So lassen sich Flecken vermeiden.

Das Entwickeln dauert nur wenige Sekunden. Das Ende erkennt man daran, daß an den belichteten Stellen das blanke Kupfer zu sehen ist (Bild 6). Während des Entwickelns empfiehlt es sich, die Plastikschale zu schwenken oder leicht zu kippen (Vorsicht, Spritzer!). So erreicht man, daß der abgelöste Lack von der Platine gewaschen wird und man jederzeit erkennen kann, wie weit die Entwicklung fortgeschritten ist. Nach der Entwicklung wird die Platine unter kaltem, fließendem Wasser abgeduscht und somit fixiert. Ab jetzt darf die Platine auf keinen Fall mehr mit den Fingern angefaßt werden. Es kann nämlich sonst passieren, daß das Kupfer durch Fett, das am Kupfer haftet, nicht sauber weggeätzt wird.

Jetzt können Sie den Entwickler ansetzen. Lösen Sie dazu den Inhalt des Ätzmittelpäckchens in der angegebenen Wassermenge. Eine Faustregel besagt, daß in einem Liter Wasser etwa 200 Gramm Eisen-III-chlorid gelöst werden sollten. Das Wasser sollte etwa 40 bis 50 Grad warm sein. So löst sich das Salz schneller auf und auch die Entwicklung geht zügiger.

Machen Sie das Wasser aber auf keinen Fall heißer, denn sonst entweicht das stechend riechende und giftige Gas Chlor. Da während des Ätzvorganges auch immer etwas Chlor frei wird, sollten Sie für eine gute Lüftung Ihres Arbeitsplatzes sorgen. Statt Eisen-III-chlorid wird häufig auch Ammoniumpersulfat verwendet.

Hüten sollten Sie sich allerdings vor einer Mischung, die gerne von Profis angesetzt wird: 9 Teile Salzsäure und 1 Teil Wasserstoffperoxid. Dieses Ätzbad setzt dermaßen viel Chlor frei und ist so aggressiv, daß man die Finger davon lassen sollte.

Bringen Sie nun, wie beim Entwickeln, die Platine wieder in das Ätzbad und kippen Sie die Schale ab und zu etwas auf und ab (Bild 7). So wird der Ätzvorgang beschleunigt. Nach etwa 5 bis 10 Minuten ist der Ätzvorgang beendet. Das vorher blanke Kupfer ist restlos von der Platine entfernt. Spülen Sie die Platine wieder unter Wasser ab und trocknen Sie sie mit etwas saugfähigem Papier. Den restlichen Photolack sollte man zum Schutz des Kupfers vor Oxidation stehen lassen. Wer den Lack trotzdem entfernen möchte, kann die Platine mit Aceton abwischen.

Der chemische Teil ist nun abgeschlossen. Jetzt wird es wieder mechanisch: Es müssen die Löcher für die Bauteile gebohrt werden. Dazu brauchen Sie eine Bohrmaschine und einen Bohrer mit 1 mm Durchmesser. Ob Sie nun eine Minibohrmaschine wie auf dem Foto verwenden, oder eine Bohrmaschine in einem Bohrständer, bleibt Ihnen überlassen. Mit der Minibohrmaschine hat man etwas mehr Gefühl. Die kleinen Bohrer brechen deshalb nicht so schnell ab. Vor allem dann, wenn man einen Bohrständer zu einer solchen Miniatur-Bohrmaschine sein eigen nennen darf.

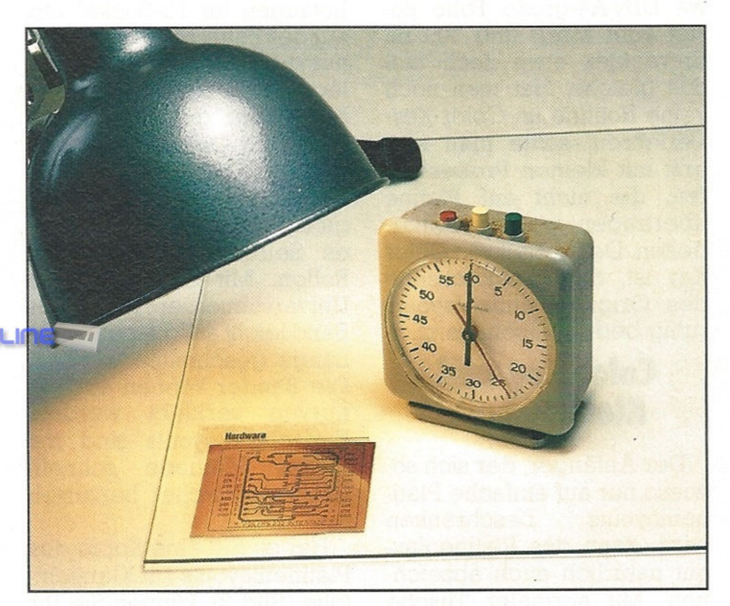


Bild 5. Die Belichtung der Platine sollte eher zu kurz als zu lang sein. Die Fassung sollte die Leistung der Birne verkraften.

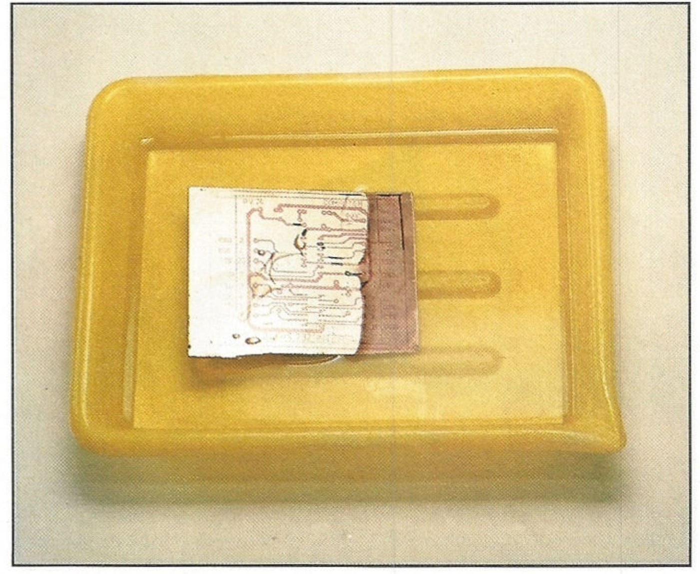


Bild 6. So sollte die belichtete und entwickelte Platine aussehen

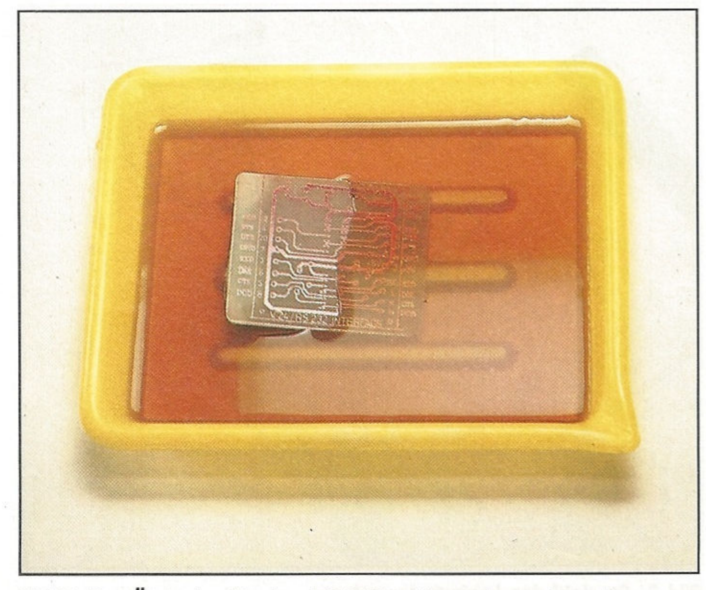


Bild 7. Das Ätzbad sollte durch leichtes Kippen bewegt werden

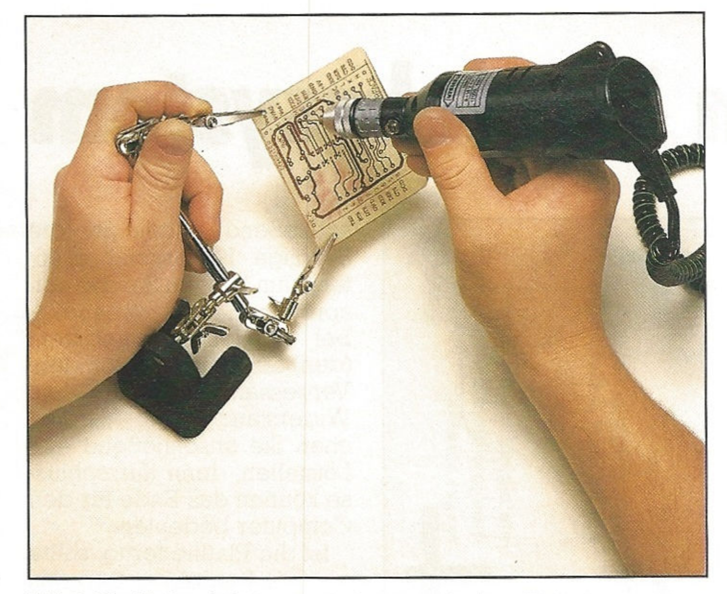


Bild 8. Die Platine bohrt man am besten mit einer Minibohrmaschine

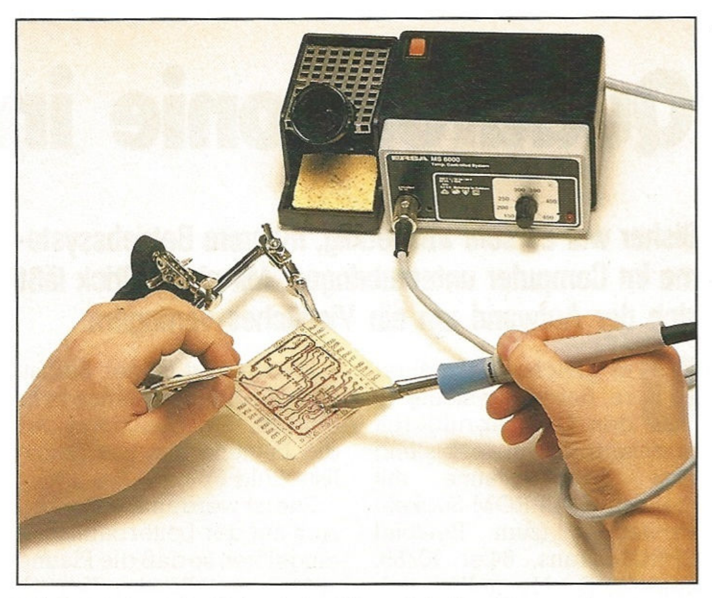


Bild 9. Die Bauteile dürfen beim Löten nicht überhitzt werden

Die Löcher bohrt man von der beschichteten Seite aus. Am besten mit einem alten Holzbrett als Unterlage. Nach dem Bohren wird eine Sichtkontrolle vorgenommen, bei der man die Platine auf feine Leiterbahnunterbrechungen und Überbrückungen hin prüfen sollte. Unterbrechungen können mit einem kleinen Drahtstück geflickt und Brücken mit einem Schraubendreher weggekratzt werden.

Hat man alle Bauteile zur Hand, kann gleich mit dem Bestücken und dem Löten begonnen werden. Aber auch hier gibt es einige Grundregeln zu beachten, die das Arbeiten erleichtern und helfen, Fehler und Ausfälle zu vermeiden. Da ist zum einen die Reihenfolge, in der die Bauteile eingelötet werden. In der Regel wird mit nicht elektronischen Bauteilen wie Sockeln und Drahtbrücken begonnen. Danach kommen die passiven Bauelemente, Widerstände, Kondensatoren, Dioden etc. an die Reihe und zum Schluß die aktiven Elemente, wie Transistoren und ICs. Man sollte sich angewöhnen, alle ICs zu sockeln. Das kostet vielleicht ein paar Mark mehr, aber man tut sich um vieles leichter, sollte einmal ein Baustein ausgewechselt werden.

Zum Löten selbst sollte man berücksichtigen, daß ein Lötkolben mit 30 Watt das Äußerste ist, was Platinen und Bauteile vertragen. Wenn die Kupferkaschierung nämlich zu heiß wird, löst sie sich ab und die Platine wird unbrauchbar. Auch die Bauteile sollten immer nur kurze Zeit in Berührung mit dem Lötkolben sein, damit sie nicht überhitzt werden. Bei Transistoren, Dioden und vor allem bei LEDs sollte die Lötwärme immer mit einer kleiner Zange oder Pinzette vor dem eigentlichen Bauteil abgeführt werden.

So wird gelötet

Als Regel sollten Sie beherzigen, daß ein heißer Lötkolben allen Bauteilen wesentlich zuträglicher ist als ein gerade aufgeheizter, mit dem man lange braten muß, bevor das Lot schmilzt.

Als Lötzinn ist ausschließlich Elektroniklot mit einem Durchmesser von maximal 1 mm geeignet. Dieses Lot hat eine Seele aus Kolophonium, das beim Schmelzen ein Oxidieren des Lots und Kupfers verhindert und als Flußmittel dient.

Am einfachsten geht das Löten, wenn der Lötkolben eine zunderfreie Spitze hat. Oxidationsreste lassen sich bei dieser Sorte leicht mit einem Stück altem Baumwollstoff abwischen. Salmiaksteine und Lötwasser sind Dinge, die zum Platinenlöten keinesfalls hergenommen werden dürfen.

Doch genug der Theorie. Stecken Sie die Bauteile auf die Platine und biegen Sie die Beine leicht schräg zur Seite ab. So verhindern Sie, daß die Bauteile aus den Löchern herausrutschen, wenn Sie die Platine zum Löten herumdrehen.

Beim Bestücken und Löten ist eine sogenannte »dritte Hand« von großem Vorteil (Bild 9). Man hat eine Hand für den Lötkolben und eine für das Lot frei. Wenn nun der Lötkolben richtig heiß ist, wird ein Bauteil-Beinchen mit der Spitze erwärmt. Und zwar so, daß die Spitze des Lötkolbens auch den Kupferring um die Bohrung erreicht. Gleichzeitig bringt man etwas Lot an die Lötstelle. Sobald sich das Lot gleichmäßig über die gesamte Lötstelle verteilt hat, wird der Lötkolben entfernt. Während des Abkühlens darf die Lötstelle nicht bewegt oder erschüttert werden, sonst kommt es zu einer »kalten« Lötstelle, die nachgelötet werden muß. Eine »kalte« Lötstelle erkennt man an ihrer matten, kristallartigen Oberfläche. Sie hat die unangenehme Eigenschaft, den Strom nur dann zu leiten, wenn es ihr paßt.

Nach dem Löten werden die abstehenden Drähte knapp über den Lötpunkten mit einem kleinen Seitenschneider abgezwickt. Die Vertiefung im Seitenschneider muß von der Platine wegzeigen, damit der Scherdruck nicht auf die Lötstelle geht.

Sind alle Bauteile eingelötet, ist noch mal eine Sichtkontrolle angesagt. Es kann nämlich leicht vorkommen, daß sich haarfeine Lötbrücken gebildet haben. Das geschieht meist bei eng beieinander liegenden Lötstellen, wie es zum Beispiel bei IC-Sockeln der Fall ist. Eine sorgfältige Kontrolle kann eine spätere, langwierige Fehlersuche ersparen.

Ist die Platine komplett fertig, sollte sie in ein Gehäuse eingebaut werden. Das erspart Kurzschlüsse und das Abbrechen von Bauteilen und sieht vor allem professioneller aus. Plastikgehäuse sind sehr leicht zu bearbeiten und billig.

Sowohl beim Platinenätzen als auch beim Löten gilt die Regel, daß noch kein Meister vom Himmel gefallen ist. Erst Übung macht den Meister.

Sollte Sie dieser Beitrag zu ersten Lötversuchen angestiftet haben, können Sie sich ein paar Lochrasterplatinen besorgen, in die Sie irgendwelche Drähte einlöten. Wenn Sie genug Erfahrung haben (geht ganz schnell!), können Sie sich ans Bestücken von Platinen wagen.

Literatur zu diesem Thema findet man in den meisten Elektronik-Läden. Aber auch zu den Produkten bekommt man gute Tips.

(Udo Reetz/hm)



Quadrophonie im Betriebssystem

Bisher war es sehr aufwendig, mehrere Betriebssysteme im Computer unterzubringen. Mit einem Trick läßt sich der Aufwand um ein Vielfaches verringern.

Um gleichzeitig mehrere Betriebssysteme in den C 64 abrufbereit einzusetzen, wurde bislang eine Adapterplatine mit mehreren EPROM-Sockeln verwendet (zum Beispiel EPROM-Trans, 64'er 10/85, Seite 42). Die Herstellung solcher Platinen ist aufgrund ihres Umfanges recht zeitaufwendig und kompliziert.

Unser neuer Kernel-Adapter besteht aus einer kleinen und sehr einfachen Platine (Bild 1). Dabei können in einem großen EPROM mehrere Betriebssysteme gleichzeitig untergebracht werden. So lassen sich in einem EPROM des Typs 27256 (32 KByte) vier, und im nächstkleineren Typ 27128 (16 KByte) zwei Betriebssysteme unterbringen. Der Preis für ein solches EPROM liegt meist wesentlich unter 30 Mark. Um die Platine so einfach wie möglich zu halten, haben wir auf Flip-Flops für die Umschaltung des Betriebssystems verzichtet. Deshalb sollte nur bei ausgeschaltetem Computer umgeschalten werden, andernfalls wäre ein Systemabsturz unumgänglich. Für ein 16-KByte-EPROM brauchen Sie einen, für das 32-KByte-EPROM zwei Umschalter (1 x Um). Bild 2 zeigt den Anschluß des 27128, in Bild 3 finden Sie die Schaltungsvariante für den 27256. Bei Verwendung des 32-KByte-EPROMs muß die Leiterbahn zwischen Anschluß 1 und 2 an der vorgesehenen Stelle unterbrochen werden (Bild 3). Bei der Verschaltung darf auf keinen Fall der 10-KOhm-Widerstand zwischen dem Schalter und der Versorgungsspannung vergessen werden.

Die Betriebssysteme werden hintereinander auf das EPROM programmiert. In den Bildern 4 und 5 wird das Schema der EPROM-Belegung dargestellt.

Wie gesagt ist der Selbstbau der Adapterplatine sehr einfach. Sie benötigen nur die Bauteile aus der Stückliste (Bild 6).

Zuerst werden die Pin-Leisten auf der Leiterbahnseite eingelötet, so daß die Platine später anstelle des Kernel-ROMs in den Computer eingesetzt werden kann. Anschließend sollten Sie die 28polige IC-Fassung einlöten. Nun können Sie das entsprechende Flachbandkabel einlöten und mit dem (den) Schalter(n) verbinden. Vergessen Sie den 10-KOhm-Widerstand nicht! Vergleichen Sie anschließend alle Lötstellen, denn Kurzschlüsse können das Ende für den Computer bedeuten.

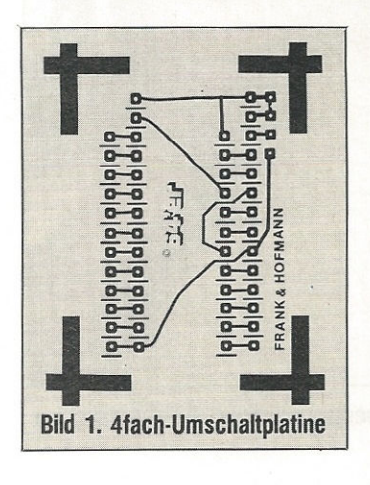


Bild 1. 4fach-Umschaltplatine

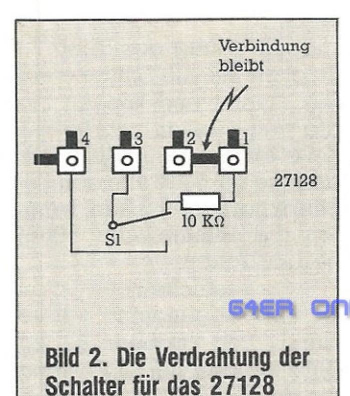


Bild 2. Die Verdrahtung der Schalter für das 27128

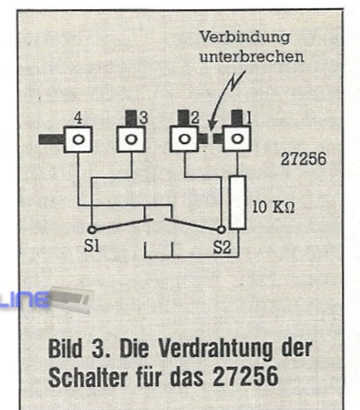


Bild 3. Die Verdrahtung der Schalter für das 27256

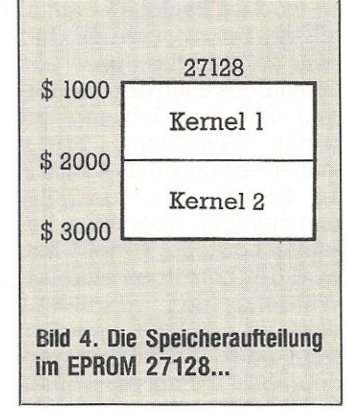


Bild 4. Die Speicheraufteilung im EPROM 27128...

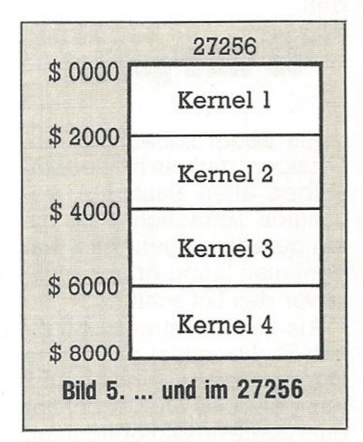


Bild 5. ... und im 27256

Zweifach-Kernel	Vierfach-Kernel
1 Platine	1 Platine (Anschluß 1 und 2 durchtrennen!)
1 Schalter (1xUm)	2 Schalter (je 1xUM)
1 IC-Fassung (28polig)	1 IC-Fassung (28polig)
2 Pin-Leisten (je 12 Pins)	2 Pin-Leisten (je 12 Pins)
1 Widerstand 10 KOhm	1 Widerstand 10 KOhm
1 EPROM 27128	1 EPROM 27256
1 3poliges Flachbandkabel	1 4poliges Flachbandkabel

Bild 6. Stückliste für die 2- beziehungsweise 4fach-Umschaltplatine

Ist die Platine fertig, sollte man die Betriebssysteme ins EPROM »brennen«. Dazu eignet sich unter anderem der 64'er EPROMmer (Ausgabe 12/85 und 1/86). Die Betriebssysteme müssen wie in Bild 4 oder 5 angeordnet werden.

Vor dem Umbau des C 64 ist unbedingt der Netzstecker zu ziehen! Lösen Sie die drei Schrauben an der Unterseite und heben Sie das Oberteil vorsichtig ab. Das Gehäuseoberteil wird nach Abstecken der Tastatur und der Leuchtdiode beiseite gelegt. Ist dies alles geschehen, ziehen Sie vorsichtig das Original-Kernel aus der Fassung (Steckplatz U4) und stecken die Platine mit der überstehenden Seite nach hinten zeigend (Richtung zum User-Port) in den C 64. Das EPROM wird ebenfalls mit der Kerbe nach hinten zeigend in die 28polige Fassung eingesetzt; die Kerbe im Sockel muß mit der Kerbe des EPROMs übereinstimmend im überstehenden Teil des Sockels liegen (Richtung zum User-Port). Die Schalter können nach Belieben am Computergehäuse befestigt werden.

Sollte Ihr Kernel-PROM eingelötet sein, lassen Sie bitte das IC von einem Fachmann durch eine 24polige Fassung ersetzen. Übernehmen Sie den Umbau selbst, so sollten Sie nach Möglichkeit eine Lötstation mit Trenntrafo verwenden.

Vorsicht! Durch Öffnen des C 64 oder durch irgendwelche Änderungen kann der Garantieanspruch verlorengehen.

(C. Q. Spitzner/M. Frank/og)

Hypra-Basic

Erstellen Sie sich Ihre eigene Befehlserweiterung mit den Befehlen, die Sie wirklich brauchen. Hypra-Basic ist eine universelle und ausbaufähige Basic-Erweiterung. Es überbietet die Mächtigkeit professioneller Erweiterungen und die Stärke teurer Toolkits bei weitem.

Hypra-Basic ist mehr als nur eine einfache Erweiterung. Durch seinen modularen Aufbau kann man in Hypra-Basic problemorientierte Erweiterungen erstellen, die die meisten kommerziellen Programme in den Schatten stellen werden. Hypra-Basic ist keine in allen Einzelheiten und Programmsegmenten feststehende Basic-Erweiterung. Vielmehr dient es zum Verknüpfen einzelner Maschinensprache-Routinen. Es ist also als Linker (aus dem englischen »to link = verbinden, verketten«) zu bezeichnen. Die Vorteile von Hypra-Basic liegen damit klar auf der Hand:

Auch mehrere Programme, die sich im Bereich ab $C000 befinden, können so gleichzeitig im Speicher liegen. Hypra-Basic verschiebt diese Programme und paßt Adressen automatisch an.

Die erstellten Routinen können im Speicher fast frei verschoben werden.

Der Aufruf der einzelnen Routinen erfolgt, wie im normalen Basic, über von Hypra-Basic frei definierbare Namen (mit Ausnahme der Basic-Befehlsworte) und nicht über SYS-Aufrufe.

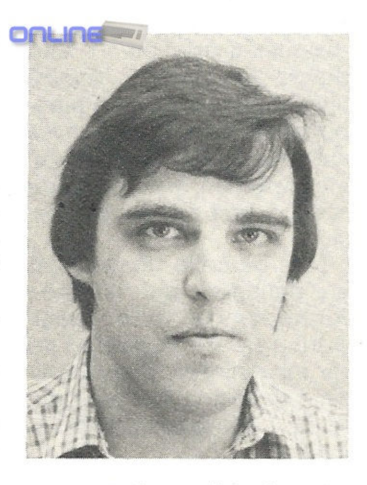
Die einzelnen Befehlserweiterungen können jeweils bestimmten Problemen angepaßt werden. Denkbar sind zum Beispiel Grafik- oder Soundprogramme genauso wie Tools oder genauere Rechenroutinen.

Im Laufe der Zeit können Sie sich eine umfassende Bibliothek anlegen, die Ihnen für alle Eventualitäten eine Lösung anbietet.

Natürlich werden wir in den nächsten Ausgaben noch viele Routinen unserer Leser abdrucken. Die vorgestellten Routinen machen den Anfang.

Ihre Aufgabe soll es sein, für Hypra-Basic Programm-Module zu schreiben. Die besten werden wir veröffentlichen.

(R. Aretz/og)



Lebenslauf

Mein Name ist Richard Aretz, ich erblickte am 12.11.1958 das Licht der Welt. Mein Geburtsort heißt Kempen. Das liegt zirka 30 km von der Stadt Aachen entfernt. Nach dem Besuch der Volksschule (1965 bis 1973) erlernte ich das Handwerk des Heizungsbauers. Im Anschluß an die Gesellenprüfung folgten sieben Jahre in meinem erlernten Beruf. In der Zwischenzeit holte ich auf dem sogenannten 2. Bildungsweg in Form von Abendkursen die mittlere Reife und die Fachhochschulreife nach. Danach, seit 1983, Studium der Elektrotechnik an der Fachhochschule Aachen.

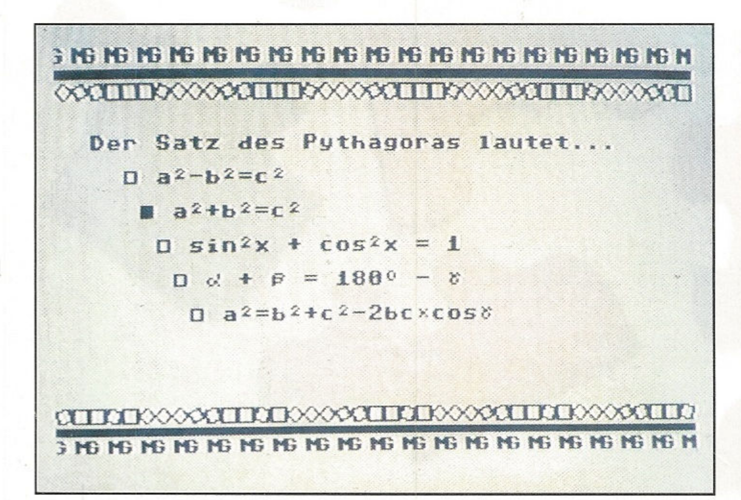
Im Verlauf des 1. Semesters kam ich zum ersten Mal mit Computern in Berührung und erlernte die Programmiersprache Fortran. Das faszinierte mich derart, daß es bis zum eigenen Computer kein weiter Weg war. Seit zirka zwei Jahren besitze ich einen C 64.

Es ist schon erstaunlich, was in einer solch »kleinen« Maschine alles steckt. Bei aller Begeisterung über diese schnell fortschreitende Technik sollte man eines nicht vergessen. Es ist eine Maschine, die keine Gefühle, Launen oder sonstige menschliche Regungen zeigen kann, sondern nur das ausführt, was ein Programmierer ihr vermittelt. Einige Filmemacher scheinen das vergessen zu haben. Solche Filme — Sie wissen sicher, über welche Art Filme ich hier spreche — tragen meines Erachtens nur zur Verunsicherung vieler Menschen bei, die nicht mit der Materie vertraut sind.

Richard Aretz

Quizmaster — die Hilfe bei Lernproblemen

Der Computer als universeller Fragebogen — Er hilft Ihnen bei Ihren Prüfungsvorbereitungen, kann aber ebenso als Party-Gag verwendet werden.



Ein typischer Multiple-Choice-Fragebogen



So kann ein selbsterstelltes Titelbild aussehen

Quizmaster ist ein Generator-Programm, das Sie befähigt, eigene Multiple-Choice-Fragen mit grafischer Untermalung zu erstellen. (Multiple Choice ist ein Fragebogen-System, bei dem man mehrere mögliche Antworten zur Auswahl hat, von denen aber nicht unbedingt alle richtig sind.) So können Sie sich zum Beispiel einen Sprachen- oder Mathematikkurs selbst erstellen und sich quasi vom Computer ausfragen lassen.

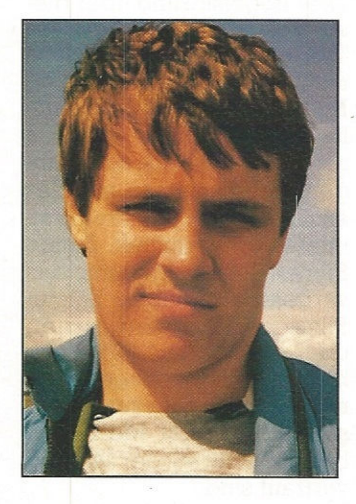
Durch die Möglichkeit, sich seine Fragen und Antworten universell, das heißt aus jedem Themengebiet zusammenzustellen und durch geschickte Anordnung und Aufbau der Fragebogenseiten ein attraktives und lehrreiches Quiz zusammenzustellen, wird das Programm erst richtig interessant.

Der aktuelle Prozentsatz an richtigen und falschen Antworten ist nach Beantwortung jeder einzelnen Frage am unteren Bildschirmrand zu sehen, was eine laufende Leistungskontrolle ermöglicht.

Die besonderen Möglichkeiten des Programms sind:
— Einbinden von bis zu zwei frei definierbaren Zeichensätzen,
— zwei Bewegungssequenzen von je 16 Zeichen,
— für jede der bis zu 99 Fragen gibt es einen eigenen Bildschirm,
— Editieren eines Bildschirms wie mit dem Basic-Editor, aber mit einigen Extras,
— alle Funktionen werden durch Menüs gesteuert.

Dieses Listing beweist wieder einmal, daß man auch mit relativ wenig Aufwand durch guten Programmierstil ein effektives Programm schreiben kann. Wenn Sie sich die Listings einmal angeschaut haben und befürchten, daß es Ihnen zuviel Tipparbeit ist, so können Sie beruhigt sein. Sie müssen nur »QUIZMASTER« und »←QUIZML« eingeben. Danach ist das Programm schon voll lauffähig. Das Listing »←Quizchar« bietet einen Demo-Zeichensatz an. Beim Eintippen können Sie alle REM-Zeilen weglassen.

(Matthias Gerloff/dm)



Lebenslauf

Vor 19 Jahren erblickte ich das Licht der Welt und vor vier Jahren das der Computerwelt. Auf einem alten CBM 4032 lernte ich von einem Freund Basic und versuchte mich dann selbst an kleinen Programmen. Während eines einjährigen Aufenthaltes in den USA als Austauschschüler hatte ich die Möglichkeit, da ich nahe am Silicon Valley wohnte, verschiedene Computer zu nutzen. Als ich wieder in Deutschland war, bekam ich von meiner Oma einen C 64 »gesponsert«, auf dem ich jetzt programmiere. Dabei verwende ich hauptsächlich Assembler, da die Geschwindigkeit von Basic, falls es die überhaupt gibt, nicht gerade überzeugend ist. Dennoch ist Basic sinnvoll und so ist der Quizmaster auch eine Mischung aus beiden. Besonders wichtig ist mir an Programmen, daß sie benutzerfreundlich sind. Im Moment stecke ich mitten im Abitur und hoffe es gut abschließen zu können. Danach strebe ich ein Studium in Informatik an, wobei ich mir über den Studiengang noch nicht schlüssig bin. Außer computern spiele ich noch gern Schach, male etwas und spiele ein wenig Orgel. Englische Filme, Theaterstücke, Zeitungen oder Bücher finden auch mein Interesse, was kein Wunder ist, nach einem Jahr in Amerika. Ehrenamtlich arbeite ich im YOUTH FOR UNTERSTANDING, damit auch andere Schüler Gelegenheit haben, ein Jahr im Ausland zu verbringen. Denn so ein Blick in andere Länder gibt einem manches mehr als ein Jahr Schule.

(Matthias Gerloff)




64ER ONLINE

Quizmaster

Prüfungsvorbereitungen oder Party-Gag — Dieser Quizgenerator ist für alles universell verwendbar.

Zuerst eine Funktionsbeschreibung: Wenn Sie »Quizmaster« gestartet haben, wird noch »←Quizml« geladen und das Hauptmenü erscheint. Falls sich ein Titelbild oder ein Zeichensatz auf der Diskette befindet, werden diese vorher geladen beziehungsweise angezeigt.

Grundsätzlich wird der Cursor mit der Taste »Cursor abwärts« nach unten und mit der Taste »Cursor rechts« nach oben gesteuert. Das Anwählen einer Funktion erfolgt dann mit RETURN. Durch Anwählen des Punktes »Farbmenü« können sämtliche Farbeinstellungen verändert werden. Dieses hat jedoch keinen Einfluß auf erstellte Bilder oder Fragen. Möchten Sie ein Quiz erstellen oder verändern, so wählen Sie bitte das »Editormenü« an. Doch Vorsicht: der Unterpunkt »Quiz anlegen« formatiert die eingelegte Diskette. Dieser Punkt ist wichtig, wenn Sie das Quiz zum ersten Mal anlegen.

Quiz erstellen

Um Fragen und Titelbilder zu erstellen, gehen Sie bitte in das Menü »Bild editieren«. Ihr Cursor ist das kleine Quadrat oben links. Sollten Sie nichts erkennen können, so können Sie mit den Funktionstasten F3, F5 und F7 die Farben des Cursors, des Rahmens und des Hintergrundes verändern. Ansonsten haben alle Tasten ihre gewohnten Funktionen. »SHIFT-CLR/HOME« löscht tatsächlich den Bildschirm. Dennoch gibt es ein paar Sonderfunktionen: Ihr Cursor kann nach oben und unten aus dem Bild bewegt werden. Er erscheint dann wieder auf der gegenüberliegenden Seite. Außerdem werden nach »INST« oder Anführungsstrichen keine lästigen Steuerzeichen mehr ausgegeben. Das Bearbeiten der untersten Zeile ist leider nicht möglich. Mit F1 gelangen Sie wieder ins »Editormenü«.

Erstellen Sie nun eine Frage, so müssen Sie die möglichen Antworten mit einem »@« kennzeichnen, welcher als kleines Quadrat sichtbar wird. Die richtige Antwort ist mit einem reversen Klammeraffen (»@«) kenntlich zu machen. Sie können dieses aber auch unterlassen und die richtige Antwort erst beim Speichern per Menüauswahl kennzeichnen. Wenn Sie nun eine Frage speichern wollen, so wählen Sie den Punkt »Bild speichern« an. Falls Sie noch keine Antwort markiert haben, können Sie dieses hier nachholen. Geben Sie nun eine Nummer ein, die nicht Null und nicht größer als die maximale Anzahl ist. Haben Sie sich vertippt, so geben Sie einfach die richtige Nummer ein und die falsche wird nach links aus der Anzeige herausgeschoben. So können Sie ein Bild auch als Titelbild von Ihrem Quiz oder vom »Quizmaster« verwenden. Dazu benutzen Sie die entsprechenden Funktionen.

Einbau von Titelbildern

Wovon hängt es denn nun ab, ob das Titelbild vom »Quizmaster« oder vom Quiz selber benutzt wird? Davon, ob es auf der »Quizmaster«-Diskette oder der mit Ihrem eigenen Quiz steht. Das bedeutet, es wird immer das Titelbild benutzt, das sich auf der im Laufwerk eingelegten Diskette befindet.

Der Zeichensatz

Nun ist der Commodore-Zeichensatz nicht gerade geeignet, um tolle Titelbilder oder Fragen mit Sonderzeichen zu erstellen. Deshalb können Sie selbstdefinierte Zeichensätze nachladen und auch speichern. Soll ein Zeichensatz von Ihrem Quiz benutzt werden, so muß er unter dem Namen »←Quizchar« auf die entsprechende Diskette gespeichert werden. Es gilt das gleiche, das auch zum Titelbild gesagt wurde. Damit Sie Ihren persönlichen Zeichensatz nutzen können, muß er folgende Bedingungen erfüllen:

— er muß als PRG-File gespeichert sein,
— er muß den Aufbau haben, der den Bildern 1 und 2 zu entnehmen ist.

Es kann auch ein Doppelzeichensatz definiert werden. Das sind zwei Zeichensätze, die hintereinander gespeichert sind. So ein Zeichensatz wäre etwa ein mit Hi-Eddi erstellter. Die Zeichen »@« und »RVS@« können bei allen Zeichensätzen nicht benutzt werden, da sie für die Menüauswahl und die Antworten gebraucht werden. Es ist also einfach, sich ein eigenes Titelbild mit eigenem Zeichensatz zu erstellen. Wem dieses zu viel Arbeit ist, der kann sich einen bereits vorhandenen Zeichensatz abtippen (Listing 3). Sollten Sie das Problem haben, Ihren Zeichensatz nur als Basic-Lader zu besitzen, dann gehen Sie wie folgt vor, um ihn für den Quizmaster nutzbar zu machen:

```
Load''←Quizml'',8,1:NEW:SYS 50266
```

Dann Ihren Zeichensatz laden und POKEn lassen. Als Startadresse des Zeichensatzes ist 40960 anzugeben:

```
POKE 828,83:POKE 829,58
A$='' NAME''
FOR I=1 TO LEN (A$):POKE 829+I,
 ASC(MID$(A$,I,1)):NEXT:POKE 2,LEN(A$)
```

Dann noch:

```
SYS 50283
```

und Ihr Zeichensatz wird unter »NAME« gespeichert.

Sie werden sich vielleicht gewundert haben, was die Funktion »Bewegung« bedeutet. Falls Sie diese Funktion angeschaltet haben, so werden zwei Bewegungssequenzen von je 16 Zeichen durchlaufen. Die Sequenzen benutzen die Zeichen mit den Bildschirmcodes 224 bis 239 und 240 bis 255. Bewegungssequenz bedeutet folgendes: Haben Sie irgendwo auf dem Bildschirm ein Zeichen stehen, dessen Bildschirmcode in eine der beiden Sequenzen fällt, so erscheinen dort nacheinander alle 16 Zeichen der betreffenden Sequenz.

Zur Entwirrung hier ein Beispiel: Nehmen wir einmal an, oben links in der Ecke steht ein Zeichen mit dem Bildschirmcode 240 und die »Bewegung«ist angeschaltet, dann erscheinen an derselben Stelle nacheinander die Zeichen mit dem Code 241, 242, 243, ..., 255. Auf 255 folgt dann wieder 240 und so weiter. Probieren wir das einmal aus. »Bewegung« anschalten und »Bild editieren« wählen. Dann den Bildschirm löschen und »RVSON« sowie das Grafikzeichen »(Commodore-Taste + A)« eintippen. Nun noch den Editor mit F1 verlassen und auf »Bild zeigen« gehen. Das Bild erscheint und das Zeichen ändert in wilder Weise seine Form.

Bleibt nur noch die Funktion »Quiz spielen« im Hauptmenü. Diese Funktion führt Sie selbständig durch alle Stufen des Spieles.

Hinweise zum Eintippen

Das Programm besteht aus einem Basic- und einem Maschinenspracheteil. Bitte geben Sie zuerst das Hauptprogramm (Listing 1) ein und speichern es ab. Anschließend tippen Sie den Maschinenspracheteil (Listing 2) mit Hilfe des MSE ein und speichern ihn ebenfalls. Das Listing 3 können Sie ebenfalls abtippen, ist aber nicht für den Programmablauf maßgeblich. Es enthält nur einen Probe-Zeichensatz. Aus Tabelle 1 können Sie die Belegung des Speichers ersehen.

(Matthias Gerloff/dm)

Adresse		Belegung
$0400	- $07FF	Zwischenspeicher Farb-RAM
$7DF6		Ende des Basic-Speichers
$8000	- $83FF	1. Bildschirm
$8400	- $87FF	2. Bildschirm
$8800	- $993F	Cursorsprite
$A000	- $A7FF	1. Zeichensatz
$A800	- $AFFF	2. Zeichensatz
$B000	- $B800	Menüzeichensatz
$C000	- $C4C8	Maschinenspracheteil

Tabelle 1. Die wichtigsten Programmadressen

```
10 REM ***      QUIZMASTER      ***           <025>
11 REM ***         VON          ***           <088>
12 REM ***       MATTHIAS       ***           <035>
13 REM ***       GERLOFF        ***           <151>
14 REM *** WEIDENBAUMSWEG 94 ***              <204>
15 REM ***   2050 HAMBURG 80    ***           <050>
16 REM *** TEL.(040) 7211517 ***              <247>
17 REM ***                      ***           <080>
18 REM ***     C 64 + 1541      ***           <156>
99 GOSUB 32005                                <001>
100 PRINT CHR$(8)CHR$(14)"{BLACK}"CHR$(147
    );:POKE 53280,10:POKE 53281,10            <104>
110 SYS 36631                                 <014>
119 IF PEEK(2)=0 THEN 149                     <157>
123 PRINT"{WHITE}LADEFEHLER":PRINT"{DOWN}N
    EUER VERSUCH MIT <{RVSON}SPACE{RVOFF}>
    ."                                        <118>
124 GOSUB 11000:IF K$=" "THEN PRINT"{CLR}"
    ;:GOTO 110                                <162>
125 GOTO 124                                  <213>
149 CLR:DIM RE(99)                            <062>
151 REM B1/2 - BILDSCHIRM AN                  <222>
152 REM BW   - BEWEGUNG                       <168>
153 REM F1/2 - SUCHSTART 1/2                  <170>
154 REM FC   - SUCHE CONTINUE                 <230>
155 REM PL   - PLOT TEXT                      <127>
157 REM ZS   - ZEICHENSATZ SPEICHERN          <170>
158 REM CH   - ZEICHENSAETZE COPY             <060>
159 REM EB   - BOX EINBLENDEN                 <172>
160 REM SA   - BILD SAVE                      <157>
161 REM LO   - BILD LADEN                     <156>
162 REM ED   - EDITOR                         <038>
163 REM SP   - SPRITE COPY                    <093>
164 REM IN   - INIT ROUTINE                   <127>
165 REM BF   - BEWEGUNGSFLAG                  <134>
166 REM E1/2 - ENDE DER BILDSCHIRME           <000>
170 REM *** SYS LIST ***                      <115>
171 B1=49152:B2=49202:BW=49291:F1=49662:F2
    =49675:FC=49391:PL=49636:BF=32247         <183>
172 CH=50266:EB=49528:SA=50066:LO=49987:ED
    =49696:SP=49561:IN=50190:ZS=50283         <152>
173 E1=33792:E2=34816                         <195>
174 TX$="{BLACK}":BL$="{WHITE}":TX=0:BL=1:
    REM*** FARB COMBO FUER MENUE              <170>
180 REM *** SUCH WERT FN ***                  <143>
181 HM=1:EM=1:FM=1:LM=1:SM=1:REM ** INIT D
    ER MENUES ***                             <119>
185 DEF FN FI(X)=PEEK(251)+256*PEEK(252)      <155>
186 FB$="{BLACK,WHITE,RED,CYAN,PURPLE,GREE
    N,BLUE,YELLOW,ORANGE,BROWN,LIG.RED,GRE
    Y 1,GREY 2,LIG.GREEN,LIG.BLUE,GREY 3}"    <222>
188 SYS IN:SYS SP:PRINT"{CLR}":SYS B2:POKE
     53280,10:POKE 53281,10:PRINT"{CLR}":P
    OKE 32247,0                               <182>
189 SYS PL,10,13,"{BLACK}BOOTING QUIZMASTE
    R..."                                     <254>
190 GOSUB 920:A$="←QUIZCHAR":AU=1:GOSUB 31
    500:SYS EB                                <188>
200 A$="←QUIZTITEL":GOSUB 31510:AU=0:IF E=
    0 THEN 300                                <171>
210 PRINT"{CLR,YELLOW,10SPACE}Q U I Z M{SH
    IFT-SPACE}A{SHIFT-SPACE}S{SHIFT-SPACE}
    T{SHIFT-SPACE}E{SHIFT-SPACE}R"            <131>
211 PRINT"{10SPACE}UUUUUUUUUUUUUUUUUUUU"      <035>
212 SYS PL,18,5,"VON"                         <202>
213 SYS PL,11,9,"MATTHIAS GERLOFF"            <252>
214 GOSUB 11000:GOTO 400                      <115>
300 GOSUB 40000:REM *** SHOW PIC ***          <240>
399 REM *** HAUPTMENUE ***                    <163>
400 PRINT TX$"{CLR}"                          <022>
401 SYS PL,15,6,BL$"HAUPTMENUE":SYS PL,15,
    7,TX$"UUUUUUUUUU"                         <074>
402 SYS PL,12,9,BL$"@"TX$" QUIZ SPIELEN"      <049>
403 SYS PL,12,11,BL$"@"TX$" EDITORMENUE"      <107>
404 SYS PL,12,13,BL$"@"TX$" FARBMENUE"        <085>
405 SYS PL,12,15,BL$"@"TX$" ENDE"             <214>
410 FS=F2:EN=E2:F=HM:GOSUB 34000:GOSUB 330
    00                                        <247>
420 HM=F:ON F GOSUB 1000,500,700,430          <136>
425 GOTO 400                                  <123>
430 PRINT"{CLR}":END                          <114>
499 REM *** EDITORMENUE ***                   <233>
500 PRINT TX$"{CLR}"                          <122>
501 SYS PL,13,0,BL$"EDITORMENUE":SYS PL,13
    ,1,TX$"UUUUUUUUUUU"                       <248>
502 SYS PL,10,4,BL$"@"TX$" BILD ZEIGEN        <043>
503 SYS PL,10,5,BL$"@"TX$" BILD EDITIEREN"    <162>
504 SYS PL,10,6,BL$"@"TX$" BILD LADEN"        <211>
505 SYS PL,10,7,BL$"@"TX$" BILD SPEICHERN"    <169>
506 SYS PL,10,8,BL$"@"TX$" ZEICHENSATZ LAD
    EN"                                       <219>
507 SYS PL,10,9,BL$"@"TX$" ZEICHENSATZ SPE
    ICHERN"                                   <072>
508 SYS PL,10,10,BL$"@"TX$" TITELBILD LADE
    N"                                        <250>
509 SYS PL,10,11,BL$"@"TX$" TITELBILD SPEI
    CHERN"                                    <050>
510 SYS PL,10,12,BL$"@"TX$" QUIZ ANLEGEN"     <071>
511 SYS PL,10,13,BL$"@"TX$" BEWEGUNG : AN
    ":IF PEEK(BF)=0 THEN SYS PL,23,13,TX$"
    AUS"                                      <058>
512 SYS PL,10,14,BL$"@"TX$" HAUPTMENUE{HOM
    E}"                                       <061>
513 F=EM:GOSUB 34000:GOSUB 33000              <250>
516 EM=F:IF F=11 THEN EM=1:RETURN             <114>
517 IF F<>3 AND F<>4 THEN POKE FN FI(0),0     <243>
518 ON F GOTO 521,520,530,535,519,522,540,
    541,551,550                               <013>
519 GOSUB 900:GOTO 500                        <186>
520 SYS ED:PRINT"{RVOFF}";:GOTO 513           <104>
521 GOSUB 40000:GOTO 513                      <238>
522 GOSUB 800:GOTO 500                        <181>
530 GOSUB 32500:X=XX:GOSUB 63500:IF A$="00
    "OR F>XX THEN GOSUB 10000:GOTO 513        <249>
531 A$="FRAGE "+A$:GOSUB 31510:GOSUB 10000
    :POKE FN FI(0),0:EM=1                     <094>
532 IF PEEK(BF)=1 THEN SYS PL,23,13,TX$"AN
    "                                         <238>
533 IF PEEK(BF)=0 THEN SYS PL,23,13,TX$"AU
    S"                                        <180>
534 GOTO 513                                  <098>
535 FS=F1:EN=E1:POKE 2,128:SYS FS:IF FN FI
    (0)<>EN THEN 538                          <021>
536 POKE 2,0:SYS FS:IF FN FI(0)=EN THEN 59
    5                                         <212>
537 SYS B1:GOSUB 33000:SYS B2:POKE FN FI(0
    ),128                                     <115>
538 FS=F2:EN=E2:GOSUB 32500:X=XX+1:GOSUB 6
    3500:IF A$="00"OR F>X THEN GOSUB 10000
    :GOTO 513                                 <136>
539 A$="FRAGE "+A$:GOSUB 31520:XX=XX-(X=F)
    :GOSUB 32600:GOSUB 10000:GOTO 513         <035>
540 SYS PL,15,24,BL$"LOADING...";:A$="←QUI
    ZTITEL":GOSUB 31510:GOSUB 10000:EM=1:G
    OTO 513                                   <079>
541 SYS PL,15,24,BL$"SAVING...";:A$="←QUIZ
    TITEL":GOSUB 31520:GOSUB 10000:GOTO 51
    3                                         <242>
550 POKE BF,1-PEEK(BF):GOTO 511               <089>
551 GOSUB 63000:IF IN$=""THEN GOSUB 10000:
    GOTO 513                                  <017>
552 GOSUB 10000:SYS PL,0,24,BL$"LEERE DISK
    ETTE EINLEGEN. <{RVSON}SPACE{RVOFF}>";    <023>
553 GOSUB 11000:GOSUB 10000:IF K$<>" "THEN
     513                                      <096>
554 SYS PL,10,24,BL$"FORMATIERE "IN$;         <236>
555 GOSUB 38000:OPEN 1,8,15,"N:"+IN$+",MG"
    :GOSUB 61000:IF E<>0 THEN GOSUB 10000:
    CLOSE 1:GOTO 513                          <134>
556 XX=0:GOSUB 32600:GOSUB 10000:PRINT#1,"
    I":CLOSE 1:GOTO 513                       <232>
595 GOSUB 10000:SYS PL,5,24,BL$"KEINE ANTW
    ORT MOEGLICH. <{RVSON}SPACE{RVOFF}>";     <181>
596 GOSUB 11000:IF K$<>" "THEN 596            <163>
597 GOSUB 10000:FS=F2:EN=E2:GOTO 513          <012>
599 REM *** FARBWAHLMENUE ***                 <134>
600 PRINT TX$"{CLR}"                          <224>
601 SYS PL,13,0,BL$"FARBWAHLMENUE":SYS PL,
    13,1,TX$"UUUUUUUUUUUUU"                   <115>
602 SYS PL,14,4,BL$"@"TX$" SCHWARZ"           <065>
603 SYS PL,14,5,BL$"@"TX$" WEISS"             <048>
604 SYS PL,14,6,BL$"@"TX$" ROT"               <060>
605 SYS PL,14,7,BL$"@"TX$" TUERKIS"           <032>
606 SYS PL,14,8,BL$"@"TX$" VIOLETT"           <135>
607 SYS PL,14,9,BL$"@"TX$" GRUEN"             <107>
608 SYS PL,14,10,BL$"@"TX$" BLAU"             <235>
609 SYS PL,14,11,BL$"@"TX$" GELB"             <146>
610 SYS PL,14,12,BL$"@"TX$" ORANGE"           <014>
611 SYS PL,14,13,BL$"@"TX$" BRAUN"            <191>
612 SYS PL,14,14,BL$"@"TX$" HELLROT"          <186>
613 SYS PL,14,15,BL$"@"TX$" DUNKELGRAU"       <203>
614 SYS PL,14,16,BL$"@"TX$" GRAU"             <077>
615 SYS PL,14,17,BL$"@"TX$" HELLGRUEN"        <205>
616 SYS PL,14,18,BL$"@"TX$" HELLBLAU"         <051>
```

```
617 SYS PL,14,19,BL$"@"TX$" HELLGRAU"           <124>
620 GOSUB 34000:GOSUB 33000:F=F-1:RETURN        <182>
699 REM *** FARBMENUE ***                       <013>
700 PRINT"{CLR}"                                <180>
701 SYS PL,15,5,BL$"FARBMENUE":SYS PL,15,6
    ,TX$"TTTTTTTTT"                             <153>
702 SYS PL,13,8,BL$"@"TX$" HINTERGRUND"         <065>
703 SYS PL,13,10,BL$"@"TX$" BLOCK U. TITEL
    "                                           <095>
704 SYS PL,13,12,BL$"@"TX$" SCHRIFT"           <234>
705 SYS PL,13,14,BL$"@"TX$" HAUPTMENUE"        <074>
710 F=FM:GOSUB 34000:GOSUB 33000                <207>
725 IF F=4 THEN FM=1:RETURN                     <037>
726 FM=F:ON F GOSUB 732,730,731                 <154>
727 GOTO 700                                    <195>
730 F=BL+1:GOSUB 600:BL=F:BL$=MID$(FB$,F+1
    ,1):RETURN                                  <079>
731 F=TX+1:GOSUB 600:TX=F:TX$=MID$(FB$,F+1
    ,1):RETURN                                  <207>
732 F=(PEEK(53280)AND 15)+1:GOSUB 600:POKE
    53280,F:POKE 53281,F:RETURN                 <092>
799 REM ** ZS - SAVER **                        <087>
800 PRINT"{CLR}":SYS PL,10,5,BL$"ZEICHENSA
    TZ SPEICHERN"                               <158>
801 SYS PL,10,6,TX$"TTTTTTTTTTTTTTTTTTTTTTT"    <145>
802 SYS PL,12,9,BL$"@"TX$" 1.ZEICHENSATZ"       <245>
804 SYS PL,12,11,BL$"@"TX$" DOPPELZEICHENS
    ATZ"                                        <212>
805 SYS PL,12,13,BL$"@"TX$" EDITORMENUE"        <009>
806 F=SM:GOSUB 34000:GOSUB 33000                <002>
810 IF F=3 THEN SM=1:RETURN                     <226>
811 SM=F:ON F GOSUB 820,822:GOSUB 63000:IF
    IN$=""THEN 806                              <061>
812 FOR I=1 TO LEN(IN$):POKE 829+I,ASC(MID
    $(IN$,I,1)):NEXT:POKE 2,LEN(IN$):SYS Z
    S                                           <170>
813 IF PEEK(2)=5 THEN GOSUB 60000:GOTO 812      <083>
814 GOSUB 61000:GOSUB 10000:GOTO 806            <142>
820 POKE 50349,168:RETURN                       <238>
822 POKE 50349,176:RETURN                       <228>
899 REM ** ZS LADEN **                          <054>
900 PRINT"{CLR}":SYS PL,11,5,BL$"ZEICHENSA
    TZ LADEN"                                   <171>
901 SYS PL,11,6,TX$"TTTTTTTTTTTTTTTTTTT"        <165>
902 SYS PL,12,9,BL$"@"TX$" 1.ZEICHENSATZ"       <089>
903 SYS PL,12,11,BL$"@"TX$" 2.ZEICHENSATZ"      <223>
904 SYS PL,12,13,BL$"@"TX$" DOPPELZEICHENS
    ATZ"                                        <064>
905 SYS PL,12,15,BL$"@"TX$" EDITORMENUE"        <175>
906 F=LM:GOSUB 34000:GOSUB 33000                <245>
910 IF F=4 THEN LM=1:RETURN                     <227>
911 LM=F:ON F GOSUB 920,921,920:GOSUB 6300
    0:IF IN$=""THEN 906                         <079>
912 A$=IN$:GOSUB 31500                          <196>
915 SYS EB:GOSUB 10000:GOTO 906                 <228>
920 POKE 36686,160:RETURN                       <066>
921 POKE 36686,168:RETURN                       <131>
999 REM *** QUIZ SPIELEN ***                    <185>
1000 GOSUB 10000:SYS PL,3,24,BL$"SPIELDISK
     ETTE EINLEGEN. <{RVSON}SPACE{RVOFF}>"
     ;                                          <130>
1005 GOSUB 11000:IF K$<>" "THEN 1005            <053>
1006 GOSUB 10000:SYS PL,13,24,TX$"BOOTING
     QUIZ...";                                  <238>
1010 GOSUB 920:AU=1:A$="←QUIZCHAR":GOSUB 3
     1500:A$="←QUIZTITEL":GOSUB 31510:AU=0      <099>
1020 GOSUB 10000:SYS EB:IF E=0 THEN GOSUB
     40000                                      <205>
1030 PRINT"{CLR}":GOSUB 32500:GOSUB 32800       <107>
1040 PRINT TX$"{CLR}":SYS PL,13,0,BL$"QUIZ
     SPIELEN":SYS PL,13,1,TX$"TTTTTTTTTTT
     T"                                         <110>
1050 SYS PL,10,5,BL$"NAME: "TX$QU$              <122>
1060 SYS PL,16,7,TX$"{LEFT}"XX"FRAGEN":IF
     XX=0 THEN GOSUB 11000:RETURN               <078>
1070 SYS PL,10,10,BL$"WIE VIELE WOLLEN SIE
     ? ";:IN$="":E=0                            <077>
1080 PRINT"T{LEFT}";:GOSUB 11000:IF K$=CHR
     $(13)THEN PRINT" ";:GOTO 1100              <072>
1090 IF E>0 AND K$=CHR$(20)THEN E=E-1:IN$=
     LEFT$(IN$,E):PRINT K$;:GOTO 1080           <183>
1095 IF K$>="0"AND K$<="9"AND E<2 THEN IN$
     =IN$+K$:PRINT K$;                          <226>
1096 GOTO 1080                                  <021>
1100 NU=VAL(IN$):SYS PL,10,10,TX$"GEWAEHLT
     E ANZAHL:"NU"{LEFT,9SPACE}";:IF NU=0
     THEN RETURN                                <049>
1101 SYS PL,10,15,TX$"FRAGEN: ":SYS PL,10,
     17,TX$"RICHTIGE ANTWORTEN:"                <080>
1102 SYS PL,10,19,TX$"FALSCHE ANTWORTEN :"      <047>
1105 FOR I=1 TO XX:RE(I)=I:NEXT                 <163>
1106 GOSUB 10000:SYS PL,5,24,BL$"{RVSON}G{
     RVOFF}EMISCHT ODER DER{SPACE,RVSON}R{
     RVOFF}EIHE NACH.";                         <079>
1107 GOSUB 11000:IF K$="R"THEN 1150             <131>
1108 IF K$<>"G"THEN 1107                        <177>
1109 GOSUB 10000                                <073>
1110 FOR I=1 TO XX                              <172>
1120 A%=RND(TI)*XX+1:B%=RND(TI)*XX+1            <008>
1130 E=RE(A%):RE(A%)=RE(B%):RE(B%)=E            <075>
1140 NEXT                                       <134>
1150 RI=0:FA=0                                  <097>
1160 FOR YY=1 TO NU                             <225>
1170 A$="FRAGE "+RIGHT$("00"+MID$(STR$(RE(
     YY)),2),2)                                 <246>
1180 GOSUB 31510:IF E<>0 THEN RETURN            <228>
1190 POKE 2,128:SYS F1:AT=FN FI(0):POKE AT
     ,0                                         <177>
1200 SYS B1:MO=1:FS=F1:EN=E1:POKE 198,0:GO
     SUB 33000:SYS B2:MO=0                      <165>
1210 IF FN FI(0)=AT THEN RI=RI+1:GOTO 1220      <082>
1211 FA=FA+1:GOSUB 10000:SYS PL,3,24,BL$"L
     EIDER FALSCH. RICHTIG WAR..<{RVSON}SP
     ACE{RVOFF}>";                              <010>
1212 POKE FN FI(0),0:POKE AT,128:GOSUB 110
     00:GOSUB 40000:GOSUB 10000                 <101>
1213 GOTO 1230                                  <009>
1220 GOSUB 10000:SYS PL,10,24,BL$"DIE ANTW
     ORT IST RICHTIG.";                         <245>
1230 SYS PL,18,15,TX$YY                         <164>
1231 SYS PL,29,17,RI:PO=RI:GOSUB 32900:SYS
     PL,33,17,"{5SPACE}";:SYS PL,35,17,PO
     $;                                         <245>
1232 SYS PL,29,19,FA:PO=FA:GOSUB 32900:SYS
     PL,33,19,"{5SPACE}";:SYS PL,35,19,PO
     $;                                         <244>
1240 NEXT YY                                    <073>
1250 GOSUB 10000:SYS PL,6,24,BL$"ENDE DES
     QUIZSPIELS. <{RVSON}SPACE{RVOFF}>";        <102>
1260 GOSUB 11000:IF K$<>" "THEN 1260            <030>
1270 RETURN                                     <056>
10000 SYS PL,0,24,"{39SPACE}";:RETURN           <126>
10002 RETURN                                    <154>
11000 GET K$:IF MO=1 AND PEEK(BF)=1 THEN S
      YS BW                                     <171>
11001 IF K$=""THEN 11000                        <093>
11010 RETURN                                    <146>
31499 REM *** ZEICHEN SATZ LOAD ***             <242>
31500 FOR I=1 TO LEN(A$):POKE 36614+I,ASC(
      MID$(A$,I,1)):NEXT:POKE 2,LEN(A$)         <142>
31501 SYS 36665:IF PEEK(2)=5 THEN GOSUB 60
      000:GOTO 31500                            <067>
31502 GOTO 61000                                <024>
31509 REM ** BILD LOAD **                       <050>
31510 FOR I=1 TO LEN(A$):POKE 829+I,ASC(MI
      D$(A$,I,1)):NEXT:POKE 2,LEN(A$)           <070>
31512 SYS LO                                    <222>
31513 IF PEEK(2)=5 THEN GOSUB 60000:GOTO 3
      1510                                      <125>
31515 GOTO 61000                                <037>
31519 REM ** BILD SAVE **                       <202>
31520 FOR I=1 TO LEN(A$):POKE 829+I,ASC(MI
      D$(A$,I,1)):NEXT:POKE 2,LEN(A$)           <016>
31522 SYS SA                                    <064>
31523 IF PEEK(2)=5 THEN GOSUB 60000:GOTO 3
      1520                                      <167>
31525 GOTO 61000                                <047>
31530 RETURN                                    <090>
31999 REM *** DATA 8F00 ***                     <216>
32000 DATA 95,81,85,73,90,77,76,95,81,85,7
      3,90,67,72,65,82,0,0,0,0,0,0,162,8        <233>
32001 DATA 160,0,32,186,255,162,0,160,143,
      169,7,32,189,255,169,0,162,0,160,192      <245>
32002 DATA 32,213,255,160,0,132,2,144,2,13
      3,2,96,162,8,160,0,32,186,255,162,7       <207>
32003 DATA 160,143,165,2,32,189,255,169,0,
      162,0,160,160,32,213,255,160,0,132,2      <051>
32004 DATA 144,2,133,2,96,-1                    <239>
32005 A=36608                                   <111>
32006 READ X:IF X<>-1 THEN POKE A,X:A=A+1:
      GOTO 32006                                <177>
32007 RETURN                                    <059>
32499 REM*** READ MAX FRAGE ***                 <144>
```

Listing 1. »Quizmaster«

```
32500 GOSUB 38000:OPEN 15,8,15,"I":OPEN 5,
      8,5,"#"                                <219>
32510 PRINT#15,"U1 5 0 18 0":PRINT#15,"B-P
       5 165"                                <071>
32515 GET#5,A$,B$:XX=VAL(A$+B$):CLOSE 5:CL
      OSE 15:GOSUB 61000:IF E<>0 THEN XX=0   <185>
32520 RETURN                                 <064>
32599 REM*** WRITE MAX FRAGE ***             <046>
32600 GOSUB 38000:OPEN 15,8,15,"I":OPEN 5,
      8,5,"#":PRINT#15,"U1 5 0 18 0"         <109>
32605 PRINT#15,"B-P 5 165":A$=LEFT$(RIGHT$
      (STR$(XX),2),1)+RIGHT$(STR$(XX),1)     <137>
32610 PRINT#5,A$;:PRINT#15,"U2 5 0 18 0":C
      LOSE 5:CLOSE 15:RETURN                 <119>
32799 REM *** READ QUIZNAME ***              <008>
32800 OPEN 15,8,15,"I":OPEN 5,8,5,"#":PRIN
      T#15,"U1 5 0 18 0":PRINT#15,"B-P 5 1
      44"                                    <008>
32810 QU$="":FOR I=1 TO 16:GET#5,K$:QU$=QU
      $+K$:NEXT:CLOSE 5:CLOSE 15:RETURN      <207>
32899 REM *** PO --> PO$ ***                 <247>
32900 PO$="":PO=INT(PO/YY*100+.5):PO$=MID$
      (STR$(PO),2)+" %"                      <201>
32910 PO$=LEFT$("{2LEFT}",LEN(PO$)-3)+PO$:
      RETURN                                 <045>
32999 REM ** MENUE AUSWAHL ** ED=ENDE:FS=F
      1 BZW. F2                              <139>
33000 POKE 2,128:SYS FS:IF FN FI(0)<>EN TH
      EN GOSUB 35000:MX=E+1:SYS FS:POKE 2,
      0:GOTO 33007                           <074>
33001 GOSUB 35000:MX=E:F=1:POKE 2,0:SYS FS
      :IF FN FI(0)=EN THEN RETURN            <103>
33006 POKE FN FI(0),128                      <130>
33007 GOSUB 11000:IF K$=CHR$(13)THEN POKE
      FN FI(0),0:RETURN                      <222>
33008 IF K$="{RIGHT}"THEN POKE FN FI(0),0:
      F=F-1-MX*(F=1):GOSUB 34000:GOTO 3300
      6                                      <179>
33010 IF K$<>"{DOWN}"THEN 33007              <028>
33015 POKE FN FI(0),0:F=F+1:SYS FC:IF FN F
      I(0)=EN THEN SYS FS:F=1                <003>
33020 GOTO 33006                             <028>
33035 .                                      <198>
34000 POKE 2,0:SYS FS:IF F=1 THEN POKE FN
      FI(0),128:RETURN                       <209>
34001 FOR I=1 TO F-1:SYS FC:NEXT:POKE FN F
      I(0),128:RETURN                        <064>
34999 REM *** FINDE MA ***                   <192>
35000 PO=PEEK(2):POKE 2,0:SYS FS:E=0         <138>
35010 IF EN<>FN FI(0)THEN E=E+1:SYS FC:GOT
      O 35010                                <022>
35020 POKE 2,PO:RETURN                       <022>
37999 REM *** TEST FLOPPY AN **              <207>
38000 A$=CHR$(0):AU=1:GOSUB 31510:AU=0:RET
      URN                                    <153>
40000 SYS B1                                 <024>
40001 IF PEEK(BF)=1 THEN SYS BW              <097>
40002 GET K$:IF K$=""THEN 40001              <067>
40003 SYS B2:RETURN                          <021>
59999 REM ** FLOPPY AUS HINWEIS **           <026>
60000 GOSUB 10000:SYS PL,10,24,BL$"FLOPPY
      ANSCHALTEN";:GOSUB 11000:RETURN        <071>
60999 REM *** ERROR KANAL ***                <032>
61000 OPEN 8,8,15:INPUT#8,E,E$,TR,SE:CLOSE
       8:IF E=0 OR AU=1 THEN RETURN          <071>
61010 GOSUB 10000:SYS PL,0,24,BL$"FEHLER :
       ";E;E$;TR;SE;:GOSUB 11000:GOSUB 100
      00:RETURN                              <086>
62999 REM *** FILE EINGABE ***               <179>
63000 GOSUB 10000:SYS PL,0,24,TX$"NAME:";    <242>
63001 IN$="":E=0                             <221>
63002 PRINT"ō{LEFT}";:GOSUB 11000:IF K$=CH
      R$(13)THEN PRINT" ";:RETURN            <190>
63003 IF K$=CHR$(20)AND E>0 THEN E=E-1:IN$
      =LEFT$(IN$,E):PRINT K$;:GOTO 63002     <154>
63004 IF E=16 OR K$=CHR$(34)OR K$<CHR$(32)
      OR(K$>CHR$(127)AND K$<CHR$(160))THEN
       63002                                 <101>
63005 PRINT K$;:E=E+1:IN$=IN$+K$:GOTO 6300
      2                                      <229>
63499 REM ** EINGABE BILD NUMMER **          <160>
63500 A$="00":GOSUB 10000:SYS PL,0,24,TX$"
      BILDNUMMER 00{2SPACE}"BL$"MAX ";       <087>
63501 F=0:PRINT RIGHT$("00"+MID$(STR$(X+(X
      =100)),2),2)TX$;                       <234>
63502 GOSUB 11000:IF K$=CHR$(13)THEN RETUR
      N                                      <246>
63503 IF K$<"0"OR K$>"9"THEN 63502           <064>
63504 F=10*(F-10*INT(F/10))+VAL(K$):A$=RIG
      HT$("00"+MID$(STR$(F),2),2)            <150>
63505 SYS PL,11,24,A$;:GOTO 63502            <034>
```

Listing 1. »Quizmaster« (Schluß). Bitte mit dem Checksummer V3 eingeben.

```
programm : ←quizml                c000 c4c4
-----------------------------------------

c000 : a9 80 8d 88 02 20 fe c0   fc
c008 : ea ea ad 20 d0 8d fc 7d   3f
c010 : ad 21 d0 8d fd 7d ad 86   c3
c018 : 02 8d fe 7d ad f8 7d 8d   04
c020 : 20 d0 ad f9 7d 8d 21 d0   bd
c028 : ad fa 7d 8d 86 02 20 64   25
c030 : c0 60 a9 84 8d 88 02 20   81
c038 : 0b c1 ea ea ad 20 d0 8d   76
c040 : f8 7d ad 21 d0 8d f9 7d   e3
c048 : ad 86 02 8d fa 7d ad fc   b7
c050 : 7d 8d 20 d0 ad fd 7d 8d   92
c058 : 21 d0 ad fe 7d 8d 86 02   8f
c060 : 20 64 c0 60 a2 04 a0 00   bb
c068 : a9 00 85 8b 85 fd a9 04   db
c070 : 85 8c a9 d8 85 fe b1 8b   ef
c078 : 48 b1 fd 91 8b 68 91 fd   89
c080 : c8 d0 f3 e6 8c e6 fe ca   1b
c088 : d0 ec 60 a9 80 85 8c a9   d5
c090 : 00 85 8b a2 04 a0 00 b1   33
c098 : 8b 85 8f 29 e0 c9 e0 d0   70
c0a0 : 13 a9 0f 25 8f 18 69 01   52
c0a8 : 29 0f 85 fe a9 f0 25 8f   70
c0b0 : 05 fe 91 8b c8 d0 e0 e6   6e
c0b8 : 8c ca d0 d9 60 a9 00 85   77
c0c0 : fb a9 80 85 fc a0 00 b1   99
c0c8 : fb c5 02 d0 0b a5 fb 8d   29
c0d0 : 10 8e a5 fc 8d 11 8e 60   8d
c0d8 : e6 fb d0 eb e6 fc a5 fc   54
c0e0 : c9 88 d0 e3 a9 ff 8d 10   8f
c0e8 : 8e a9 87 8d 11 8e 60 ad   41
c0f0 : 10 8e 85 fb ad 11 8e 85   d1
c0f8 : fc a0 00 4c d8 c0 ad 18   48
c100 : d0 8d ff 7d ad fb 7d 8d   12
c108 : 18 d0 60 ad 18 d0 8d fb   8c
c110 : 7d ad ff 7d 8d 18 d0 60   b1
c118 : ad 0e dc 29 fe 8d 0e dc   77
c120 : a5 01 29 fa 85 01 a2 10   fb
c128 : a0 00 a9 00 85 fb 85 fd   7d
c130 : a9 d0 85 fc a9 b0 85 fe   76
c138 : b1 fb 91 fd c8 d0 f9 e6   d4
c140 : fe e6 fc ca d0 f2 a5 01   87
c148 : 09 05 85 01 ad 0e dc 09   26
c150 : 01 8d 0e dc 60 a0 07 b9   d2
c158 : 89 c1 91 fb 88 10 f8 a5   de
c160 : fc 18 69 04 85 fc a0 07   14
c168 : b9 91 c1 91 fb 88 10 f8   c3
c170 : 18 a5 fc 69 04 85 fc 60   e8
c178 : a9 00 85 fb a9 a0 85 fc   b2
c180 : a2 04 20 55 c1 ca d0 fa   83
c188 : 60 00 7e 42 42 42 42 7e   0c
c190 : 00 00 7e 7e 7e 7e 7e 7e   d3
c198 : 00 a0 3f b9 a5 c1 99 00   be
c1a0 : 88 88 10 f7 60 ff c0 00   78
c1a8 : 80 40 00 80 40 00 80 40   df
c1b0 : 00 80 40 00 80 40 00 80   0c
c1b8 : 40 00 80 40 00 80 40 00   26
c1c0 : ff c0 00 00 00 00 00 00   20
c1c8 : 00 00 00 00 00 00 00 00   c9
c1d0 : 00 00 00 00 00 00 00 00   d1
c1d8 : 00 00 00 00 00 00 00 00   d9
c1e0 : 00 00 00 00 20 fd ae 20   cd
c1e8 : 9e b7 8a 48 20 fd ae 20   fa
c1f0 : 9e b7 68 a8 18 20 f0 ff   df
c1f8 : 20 fd ae 4c a4 aa a9 84   9b
c200 : 8d e1 c0 a9 83 8d ea c0   b5
c208 : 4c bd c0 a9 88 8d e1 c0   96
c210 : a9 87 8d ea c0 a9 00 85   a2
c218 : fb a9 84 85 fc 4c c5 c0   84
c220 : a9 20 8d f8 83 20 00 c0   16
c228 : a9 13 20 d2 ff a9 09 20   6f
c230 : d2 ff a9 01 8d 15 d0 a9   a4
c238 : 00 8d 1c d0 8d 17 d0 8d   10
c240 : 1d d0 38 20 f0 ff 84 fb   f0
c248 : 86 fc 98 38 e9 28 90 02   a0
c250 : 85 fb a9 31 a6 fc f0 07   87
c258 : 18 69 08 ca 4c 56 c2 8d   1e
c260 : 01 d0 8e 10 d0 a9 17 a6   73
c268 : fb f0 0e 18 69 08 90 05   85
c270 : a0 01 8d 10 d0 ca 4c 69   5e
c278 : c2 8d 00 d0 ad 01 d0 c9   d5
c280 : f1 d0 08 a4 fb a6 fc ca   ee
c288 : 4c 36 c3 20 e4 ff f0 fb   ee
c290 : c9 85 d0 0d a9 08 20 d2   f3
c298 : ff a9 00 8d 15 d0 4c 32   8b
c2a0 : c0 c9 86 d0 06 ee 27 d0   17
c2a8 : 4c 8b c2 c9 87 d0 06 ee   99
c2b0 : 20 d0 4c 8b c2 c9 88 d0   fb
c2b8 : 06 ee 21 d0 4c 8b c2 c9   57
c2c0 : 94 d0 26 20 d2 ff a0 29   4c
c2c8 : ad 21 d0 99 c0 db 88 10   9a
c2d0 : f7 a2 18 a0 01 18 20 f0   66
c2d8 : ff a9 00 85 d8 a9 14 20   c8
c2e0 : d2 ff a6 fc a4 fb 4c 36   c3
c2e8 : c3 c9 11 d0 0d a6 fc e0   aa
c2f0 : 17 d0 07 a2 00 a4 fb 4c   33
c2f8 : 36 c3 c9 91 d0 0b a6 fc   ae
c300 : d0 07 a2 17 a4 fb 4c 36   a7
c308 : c3 c9 1d d0 0d a6 fb e0   c9
c310 : 27 d0 07 a0 00 a6 fc 4c   37
c318 : 36 c3 c9 9d d0 0b a6 fb   4e
c320 : d0 07 a0 27 a6 fc 4c 36   71
c328 : c3 c9 0d f0 04 c9 8d d0   98
c330 : 0c a6 fc a0 00 e8 18 20   ca
c338 : f0 ff 4c 42 c2 20 d2 ff   fc
c340 : 4c bd c4 a5 02 a2 3e a0   c0
c348 : 03 20 bd ff a2 08 a0 00   b8
c350 : 20 ba ff a9 00 a2 f7 a0   39
c358 : 7d 20 d5 ff 90 03 85 02   96
c360 : 60 a9 7e 85 8c a9 04 85   16
c368 : fc a9 06 85 fe a9 00 85   b3
c370 : 8b 85 fb 85 fd a2 02 a0   ac
c378 : 00 b1 8b 91 fb 4a 4a 4a   36
c380 : 4a 91 fd c8 d0 f3 e6 8c   8d
c388 : e6 fc e6 fe ca d0 ea 86   72
c390 : 02 60 a6 02 e8 e8 8a a2   f1
c398 : 3c a0 03 20 bd ff a9 01   6e
c3a0 : a2 08 a0 0f 20 ba ff 20   68
c3a8 : c0 ff 20 cc ff a9 01 20   9b
c3b0 : c3 ff a9 7e 85 8c a9 04   18
c3b8 : 85 fc a9 06 85 fe a9 00   dd
c3c0 : 85 8b 85 fb 85 fd a2 02   c2
c3c8 : a0 00 b1 fb 29 0f 91 fb   9d
c3d0 : b1 fd 0a 0a 0a 0a 11 fb   71
c3d8 : 91 8b c8 d0 ed e6 8c e6   91
c3e0 : fc e6 fe ca d0 e4 a2 08   37
c3e8 : 20 ba ff a5 02 a2 3e a0   89
```

```
c3f0 : 03 20 bd ff a9 f7 85 fb   db
c3f8 : a9 7d 85 fc a9 fb a2 c0   e7
c400 : a0 83 20 d8 ff a0 00 84   93
c408 : 02 90 02 85 02 60 ad 02   61
c410 : dd 09 03 8d 02 dd ad 00   aa
c418 : dd 29 fd 8d 00 dd a9 09   62
c420 : 8d 18 d0 a9 1f 8d ff 7d   7c
c428 : a9 80 8d 88 02 8d 8a 02   40
c430 : a9 7d 85 38 a9 f6 85 37   d7
c438 : a9 53 8d 3c 03 a9 3a 8d   f7
c440 : 3d 03 a9 00 85 fb a9 80   49
c448 : 85 fc a9 a0 a2 08 a0 00   b7
c450 : 91 fb c8 d0 fb e6 fc ca   ab
c458 : d0 f6 a9 a0 8d 35 c1 20   eb
c460 : 18 c1 a9 b0 8d 35 c1 20   a3
c468 : 18 c1 60 a5 01 48 a9 36   93
c470 : 85 01 a6 02 e8 e8 8a a2   a5
c478 : 3c a0 03 20 bd ff a9 01   4e
c480 : a2 08 a0 0f 20 ba ff 20   48
c488 : c0 ff 20 cc ff a9 01 20   7b
c490 : c3 ff a2 08 20 ba ff a5   20
c498 : 02 a2 3e a0 03 20 bd ff   b7
c4a0 : a9 00 85 fb a9 a0 85 fc   da
c4a8 : a9 fb a2 00 a0 a8 20 d8   79
c4b0 : ff a0 00 84 02 90 02 85   48
c4b8 : 02 68 85 01 60 a9 00 85   ce
c4c0 : d4 4c 42 c2 00 51 5f 00   ab
```

Listing 2. »←Quizml«. Bitte mit dem MSE eingeben.

```
 programm : ←quizchar             a000 a800
--------------------------------------------

a000 : 00 7e 42 42 42 42 7e 00   48
a008 : 18 3c 66 7e 66 66 66 00   db
a010 : 7c 66 66 7c 66 66 7c 00   74
a018 : 3c 66 60 60 60 66 3c 00   d6
a020 : 78 6c 66 66 66 6c 78 00   e0
a028 : 7e 60 60 78 60 60 7e 00   00
a030 : 7e 60 60 78 60 60 60 00   90
a038 : 3c 66 60 6e 66 66 3c 00   18
a040 : 66 66 66 7e 66 66 66 00   76
a048 : 3c 18 18 18 18 18 3c 00   cd
a050 : 1e 0c 0c 0c 0c 6c 38 00   fe
a058 : 66 6c 78 70 78 6c 66 00   a5
a060 : 60 60 60 60 60 60 7e 00   17
a068 : 63 77 7f 6b 63 63 63 00   b3
a070 : 66 76 7e 7e 6e 66 66 00   34
a078 : 3c 66 66 66 66 66 3c 00   d8
a080 : 7c 66 66 7c 60 60 60 00   e3
a088 : 3c 66 66 66 66 3c 0e 00   de
a090 : 7c 66 66 7c 78 6c 66 00   ed
a098 : 3c 66 60 3c 06 66 3c 00   2c
a0a0 : 7e 18 18 18 18 18 18 00   d6
a0a8 : 66 66 66 66 66 66 3c 00   32
a0b0 : 66 66 66 66 66 3c 18 00   58
a0b8 : 63 63 63 6b 7f 77 63 00   54
a0c0 : 66 66 3c 18 3c 66 66 00   fc
a0c8 : 66 66 66 3c 18 18 18 00   25
a0d0 : 7e 06 0c 18 30 60 7e 00   57
a0d8 : 3c 30 30 30 30 30 3c 00   b4
a0e0 : 0c 12 30 7c 30 62 fc 00   9b
a0e8 : 3c 0c 0c 0c 0c 0c 3c 00   c1
a0f0 : 00 18 3c 7e 18 18 18 18   ae
a0f8 : 00 10 30 7f 7f 30 10 00   b6
a100 : 00 00 00 00 00 00 00 00   01
a108 : 18 18 18 18 00 00 18 00   96
a110 : 66 66 66 00 00 00 00 00   43
a118 : 66 66 ff 66 ff 66 66 00   4b
a120 : 18 3e 60 3c 06 7c 18 00   9c
a128 : 62 66 0c 18 30 66 46 00   13
a130 : 3c 66 3c 38 67 66 3f 00   5c
a138 : 06 0c 18 00 00 00 00 00   4b
a140 : 0c 18 30 30 30 18 0c 00   5e
a148 : 30 18 0c 0c 0c 18 30 00   4b
a150 : 00 66 3c ff 3c 66 00 00   89
a158 : 00 18 18 7e 18 18 00 00   7d
a160 : 00 00 00 00 00 18 18 30   e2
a168 : 00 00 00 7e 00 00 00 00   38
a170 : 00 00 00 00 00 18 18 00   92
a178 : 00 03 06 0c 18 30 60 00   82
a180 : 3c 66 6e 76 66 66 3c 00   e4
a188 : 18 18 38 18 18 18 7e 00   fa
a190 : 3c 66 06 0c 30 60 7e 00   02
a198 : 3c 66 06 1c 06 66 3c 00   91
a1a0 : 06 0e 1e 66 7f 06 06 00   42
a1a8 : 7e 60 7c 06 06 66 3c 00   bb
a1b0 : 3c 66 60 7c 66 66 3c 00   52
a1b8 : 7e 66 0c 18 18 18 18 00   12
a1c0 : 3c 66 66 3c 66 66 3c 00   db
a1c8 : 3c 66 66 3e 06 66 3c 00   1d
a1d0 : 00 00 18 00 00 18 00 00   97
a1d8 : 00 00 18 00 00 18 18 30   60
a1e0 : 0e 18 30 60 30 18 0e 00   0e
a1e8 : 00 00 7e 00 7e 00 00 00   70
a1f0 : 70 18 0c 06 0c 18 70 00   74
a1f8 : 3c 66 06 0c 18 00 18 00   4c
a200 : 00 00 01 03 07 0f 0f 0f   e4
a208 : 00 f0 fc fc fe fe be 3e   be
a210 : 00 0f 3f 3f 7f 7c 7f 3f   a8
a218 : 00 f0 fc fc fe 3e fe fe   4b
a220 : 00 0f 3f 3f 7f 7c 7f 3f   b8
a228 : 00 f0 fc fc fe 3e fe fe   5b
a230 : 00 7f 7f 7f 7f 7c 7f 3f   18
a238 : 00 fc fe fe fc 00 f0 fc   e3
a240 : 00 7f 7f 7f 7f 78 78 78   5e
a248 : 00 f0 fc fc fe 3e 1e 1e   36
a250 : 00 0f 3f 3f 7f 7c 78 78   3e
a258 : 00 f0 fc fc fe 3e 1e 1e   46
a260 : 00 0f 3f 3f 7f 7c 7c 7f   6c
a268 : 00 fc fe fe fc 00 c0 d7   d7
a270 : 00 38 7c 7c 7c 7c 7c 7c   d2
a278 : 00 1c 3e 3e 3e 3e 3e 3e   29
a280 : 00 00 01 03 07 0f 0f 0f   64
a288 : 00 60 f0 e0 c0 80 80 80   24
a290 : 00 06 0f 07 03 01 01 01   76
a298 : 00 00 80 c0 e0 f0 f0 f0   0c
a2a0 : 60 60 60 60 60 60 60 60   a0
a2a8 : 00 00 00 07 0f 1c 18 18   ec
a2b0 : c3 e7 7e 3c 3c 7e e7 c3   6d
a2b8 : 00 3c 7e 66 66 7e 3c 00   8e
a2c0 : 18 18 66 66 18 18 3c 00   7e
a2c8 : 06 06 06 06 06 06 06 06   c8
a2d0 : 08 1c 3e 7f 3e 1c 08 00   4b
a2d8 : 18 18 18 ff ff 18 18 18   54
a2e0 : c0 c0 30 30 c0 c0 30 30   46
a2e8 : 18 18 18 18 18 18 18 18   e8
a2f0 : 00 00 03 3e 76 36 36 00   6b
a2f8 : ff 7f 3f 1f 0f 07 03 01   a2
a300 : 06 00 00 00 00 00 00 00   07
a308 : 3e 3e 3e 3e 3e 1c 00 00   03
a310 : 3f 0f 00 1f 3f 3f 1f 00   25
a318 : fe fe 3e fe fc fc f0 00   80
a320 : 7f 7f 7c 7f 3f 3f 0f 00   98
a328 : fc fe 3e fe fc fc f0 00   8e
a330 : 3f 0f 00 1f 3f 3f 0f 00   05
a338 : fc fe 3e fe fc fc f0 00   9e
a340 : 78 78 78 7f 7f 7f 7f 00   f4
a348 : 1e 1e 3e fe fc fc f0 00   60
a350 : 78 78 7c 7f 3f 3f 0f 00   3e
a358 : 0e 36 78 7c be de ce 00   4d
a360 : 7f 7f 7f 7c 7c 7c 38 00   9b
a368 : e0 e0 c0 00 00 00 00 00   e9
a370 : 7c 7e 7e 7f 7f 3f 0f 00   e8
a378 : 3e 3e 7e fe fe fc f0 00   f0
a380 : 0f 0f 0f 07 03 01 00 00   f4
a388 : 80 80 80 c0 e0 f0 60 00   98
a390 : 01 01 01 03 07 0f 06 00   b4
a398 : f0 f0 f0 e0 c0 80 00 00   69
a3a0 : c0 c0 c0 c0 c0 c0 c0 c0   9f
a3a8 : e0 e0 e0 e0 e0 e0 e0 e0   a7
a3b0 : 07 07 07 07 07 07 07 07   b0
a3b8 : ff ff 00 00 00 00 00 00   b7
a3c0 : ff ff ff 00 00 00 00 00   bf
a3c8 : 00 00 00 00 00 ff ff ff   c8
a3d0 : 00 00 00 00 00 00 00 00   d1
a3d8 : 00 00 00 00 f0 f0 f0 f0   15
a3e0 : 0f 0f 0f 0f 00 00 00 00   1d
a3e8 : 18 18 18 f8 f8 00 00 00   c1
a3f0 : f0 f0 f0 f0 00 00 00 00   b3
a3f8 : f0 f0 f0 f0 0f 0f 0f 0f   7e
a400 : 00 7e 7e 7e 7e 7e 7e 00   85
a408 : e7 c3 99 81 99 99 99 ff   34
a410 : 83 99 99 83 99 99 83 ff   ab
a418 : c3 99 9f 9f 9f 99 c3 ff   59
a420 : 87 93 99 99 99 93 87 ff   5f
a428 : 81 9f 9f 87 9f 9f 81 ff   4f
a430 : 81 9f 9f 87 9f 9f 9f ff   cf
a438 : c3 99 9f 91 99 99 c3 ff   57
a440 : 99 99 99 81 99 99 99 ff   09
a448 : c3 e7 e7 e7 e7 e7 c3 ff   c2
a450 : e1 f3 f3 f3 f3 93 c7 ff   a1
a458 : 99 93 87 8f 87 93 99 ff   0a
a460 : 9f 9f 9f 9f 9f 9f 81 ff   a8
a468 : 9c 88 80 94 9c 9c 9c ff   1c
a470 : 99 89 81 81 91 99 99 ff   ab
a478 : c3 99 99 99 99 99 c3 ff   17
a480 : 83 99 99 83 9f 9f 9f ff   1c
a488 : c3 99 99 99 99 c3 f1 ff   31
a490 : 83 99 99 83 87 93 99 ff   32
a498 : c3 99 9f c3 f9 99 c3 ff   04
a4a0 : 81 e7 e7 e7 e7 e7 e7 ff   69
a4a8 : 99 99 99 99 99 99 c3 ff   1d
a4b0 : 99 99 99 99 99 c3 e7 ff   07
a4b8 : 9c 9c 9c 94 80 88 9c ff   1b
a4c0 : 99 99 c3 e7 c3 99 99 ff   83
a4c8 : 99 99 99 c3 e7 e7 e7 ff   6a
a4d0 : 81 f9 f3 e7 cf 9f 81 ff   48
a4d8 : c3 cf cf cf cf cf c3 ff   fb
a4e0 : f3 ed cf 83 cf 9d 03 ff   24
a4e8 : c3 f3 f3 f3 f3 f3 c3 ff   0e
a4f0 : ff e7 c3 81 e7 e7 e7 e7   31
a4f8 : ff ef cf 80 80 cf ef ff   39
a500 : 00 38 7c 7c 7c 7d 7d 7d   70
a508 : 00 1c 3e 3e 3e be be be   c0
a510 : 00 0f 3f 3f 7f 7c 7c 7f   1c
a518 : 00 f0 fc fc fe 3e 3e fe   48
a520 : 00 38 7c 7c 7c 7c 7c 7f   88
a528 : 00 1c 3e 3e 3e 3e 3e fe   5b
a530 : 00 0f 3f 3f 7f 7c 7c 7f   3c
a538 : 00 f0 f8 fc fc 3c 78 f0   03
a540 : 00 0f 3f 7f 7f 78 7f 7f   40
a548 : 00 fc fe fe fc 00 f8 fc   13
a550 : 00 3f 7f 7f 3f 07 07 07   16
a558 : 00 fc fe fe fc e0 e0 e0   92
a560 : 00 3f 7f 7f 3f 00 00 07   d2
a568 : 00 f0 fc fe fe 1e fc f8   c6
a570 : 00 38 7c 7c 7c 7c 7c 7c   d2
a578 : 00 00 00 00 00 00 00 00   79
a580 : 00 03 07 07 07 07 07 07   77
a588 : 00 c0 e0 e0 e0 e0 e0 e0   97
a590 : 00 0e 3f 3b 7b 79 7d 7d   43
a598 : 00 70 fc dc de 9e be be   06
a5a0 : 00 0f 3f 3f 7f 7c 78 78   8e
a5a8 : 00 f0 fc fc fe 3e 1e 1e   96
a5b0 : 00 0e 3f 3b 7b 79 7d 7d   63
a5b8 : 00 1c 3e 3e 3e be be be   70
a5c0 : 00 0f 3f 3f 7f 7c 7c 7c   c6
a5c8 : 00 fc fe fe fc 00 00 7c   ae
a5d0 : 00 0f 3f 3f 7f 7c 7f 7f   e8
a5d8 : 00 fc fe fe fc 00 c0 e0   8a
a5e0 : 00 38 7c 7c 7c 7c 3c 3c   c0
a5e8 : 00 1c 3e 3e 3e 3e 3c 3c   8d
a5f0 : 00 07 0f 18 30 63 67 66   c3
a5f8 : 00 e0 f0 18 0c e6 e6 06   47
a600 : 7d 7d 79 7b 3b 3f 0e 00   f0
a608 : be be 9e de dc fc 70 00   20
a610 : 7f 7f 7f 7c 7c 7c 38 00   4b
a618 : fe fe fe 3e 3e 3e 1c 00   63
a620 : 7f 7f 7f 7c 7c 7c 38 00   5b
a628 : fe fe fe 3e 3e 3e 1c 00   73
a630 : 7f 7f 7f 7c 7c 7c 38 00   6b
a638 : f0 f8 fc 3c 3e 3e 1c 00   b1
a640 : 3f 1f 00 3f 7f 7f 3f 00   e8
a648 : fe fe 1e fe fe fc f8 00   e8
a650 : 07 07 07 07 07 07 03 00   32
a658 : e0 e0 e0 e0 e0 e0 c0 00   15
a660 : 1f 3f 78 7f 7f 3f 0f 00   5b
a668 : e0 00 00 fc fe fe fc 00   c4
a670 : 7c 7c 7e 7f 7f 3f 0f 00   e8
a678 : 00 00 00 fc fe fe fc 00   f4
a680 : 07 07 07 07 07 07 03 00   62
a688 : e0 e0 e0 e0 e0 e0 c0 00   45
a690 : 7d 7d 7d 7c 7c 7c 38 00   47
a698 : be be be 3e 3e 3e 1c 00   73
a6a0 : 78 78 7c 7f 3f 3f 0f 00   8e
a6a8 : 1e 1e 3e fe fc fc f0 00   c0
a6b0 : 7d 7d 7d 7c 7c 7c 38 00   67
a6b8 : be be 9e de dc fc 70 00   d0
a6c0 : 7c 7c 7c 7f 7f 3f 1f 00   f8
a6c8 : 7e 3e 3e fe fe fc f8 00   90
a6d0 : 7f 7f 7c 7f 3f 3f 0f 00   48
a6d8 : e0 c0 00 fc fe fe fc 00   94
a6e0 : 3e 1e 1f 0f 0f 07 03 00   0c
a6e8 : 7c 78 f8 f0 f0 e0 c0 00   16
a6f0 : 66 67 63 30 18 0f 07 00   ff
a6f8 : 06 e6 e6 0c 18 f0 e0 00   39
a700 : 00 00 00 00 00 00 00 00   01
a708 : 00 00 00 18 18 00 00 00   8d
a710 : 00 00 00 18 18 00 00 00   95
a718 : 00 00 18 3c 3c 18 00 00   2b
a720 : 00 00 18 3c 3c 18 00 00   33
a728 : 00 18 3c 7e 7e 3c 18 00   3d
a730 : 00 18 3c 7e 7e 3c 18 00   45
a738 : 18 3c 7e ff ff 7e 3c 18   23
a740 : 18 3c 7e ff ff 7e 3c 18   2b
a748 : 18 3c 7e e7 e7 7e 3c 18   ae
a750 : 18 3c 7e e7 e7 7e 3c 18   b6
a758 : 18 7e 66 c3 c3 66 7e 18   5b
a760 : 18 7e 66 c3 c3 66 7e 18   63
a768 : 81 5a 24 42 42 24 5a 81   19
a770 : 81 42 24 00 00 24 42 81   49
a778 : 81 42 00 00 00 00 42 81   27
a780 : e0 f0 38 18 18 1c 0f 07   96
a788 : f0 f8 1c 0c 0c 0e 07 03   d0
a790 : f8 fc 0e 06 06 07 03 01   f1
a798 : fc fe 87 03 03 03 01 00   a2
a7a0 : 7e ff c3 81 81 81 00 00   63
a7a8 : 3f 7f e1 c0 c0 c0 80 00   4b
a7b0 : 1f 3f 70 60 60 e0 c0 80   a8
a7b8 : 0f 1f 38 30 30 70 e0 c0   f7
a7c0 : 07 0f 1c 18 18 38 f0 e0   22
a7c8 : 03 07 0e 0c 0c 1c f8 f0   bb
a7d0 : 01 03 07 06 06 0e fc f8   8c
a7d8 : 00 01 03 03 03 87 fe fc   dc
a7e0 : 00 00 81 81 81 c3 ff 7e   a4
a7e8 : 00 80 c0 c0 c0 e1 7f 3f   08
a7f0 : 80 c0 e0 60 60 70 3f 1f   d9
a7f8 : c0 e0 70 30 30 38 1f 0f   aa
```

Listing 3. »←Quizschar«. Dieses Listing muß nicht unbedingt abgetippt werden.

Hypra-Basic

Stricken Sie sich Ihre persönliche Erweiterung mit Hypra-Basic. Ohne überflüssigen Ballast mitzuführen, können Sie sich mit Hypra-Basic die Erweiterungen selbst herstellen, deren Befehle, Mächtigkeit und Umfang Sie selbst bestimmen können.

Geben Sie zunächst das Programm »BASIC .ASS« (Listing 1) mit dem MSE ein und speichern Sie es auf Diskette. Beachten Sie bitte das Leerzeichen vor dem Punkt. Dann geben Sie »HYPRA-BASIC« (Listing 2) sowie »DOKUMENT« (Listing 3) und »MG SPEICHER« (Listing 4) mit dem Checksummer ein und speichern diese ebenfalls.

Die Programme haben folgende Funktionen.

»MG SPEICHER« speichert ein Maschinenprogramm nach Angabe von Modulnummer, Start- und Endadresse.

»DOKUMENT« erstellt eine Auflistung der auf Diskette befindlichen Module mit allen wesentlichen Daten.

»HYPRA-BASIC« ist das Herz des Programms, das das menügesteuerte Verbinden der Module übernimmt.

»BASIC .ASS« enthält zum einen die Umrechnungsroutine, um Maschinenprogramme an andere Bereiche anzupassen. Dahinter befindet sich die Erweiterung für den Interpreter.

Eigene Programme einbinden

Wenn Sie eine eigene Routine in Hypra-Basic einbauen wollen, muß Ihr Unterprogramm folgende Voraussetzungen erfüllen: Start- und Endadresse müssen bekannt sein. Datenbytes, zum Beispiel für eine Fehlermeldung, müssen nach dem Maschinenprogrammteil stehen, also daran angehängt sein. Die Anzahl der Datenbytes muß ebenfalls bekannt sein. Die Befehlsworte der Module dürfen keine Basic-Anweisungen oder Funktionen enthalten. Verbotene Namen sind zum Beispiel »SPRITEON« (enthält das ON aus ON X GOTO/GOSUB) oder »PRINTAT« (enthält PRINT). Module, die mehrere Befehle beinhalten, müssen in sich geschlossen sein. Es dürfen maximal 16 Befehle in einem Modul enthalten sein. Ein Filename muß folgendes Format haben: "xERW . ASS".

Dabei entspricht x der Nummmer des Moduls. Beispiele für Namen sind »4ERW .ASS«, »38ERW .ASS« etc. Beachten Sie im Namen bitte wieder das Leerzeichen vor dem Punkt.

Da eine Diskette maximal 144 Einträge im Directory aufnehmen kann, und für jedes Modul zwei Einträge benötigt werden, sind auf einer Diskette 70 Module das Maximum. Bei höheren Modulnummern wird vom Computer eine andere Diskette verlangt. Bei den Modulnummern von 71 bis 140 also Disk 2, von 141 bis 210 Disk 3 etc.

Die Lage der erstellten Erweiterung im C 64: Die Befehlstabelle und die Interpreter-Erweiterung können nur im RAM-Bereich liegen, der ständig aktiv ist und nicht zu weit unten im Basic-Bereich liegt (dort liegt ja das Hypra-Basic). Diese Bereiche sind:
13200 ($3390) bis 40959 ($9FFF) und
49152 ($C000) bis 53247 ($CFFF).

Die Befehlsmodule selbst können auch im RAM unter dem ROM liegen. Dadurch erschließt sich ein Bereich von:
13200 ($3390) bis 53247 ($CFFF) und
57344 ($E000) bis 65535 ($FFFF).

Natürlich dürfen sich die beiden Bereiche nicht überlappen. Deshalb ist es vorteilhaft, wenn die Befehlsmodule hinter der Befehlstabelle liegen.

Starten des Programms: Laden Sie »HYPRA-BASIC« und starten Sie es. Das Maschinenprogramm »BASIC .ASS« wird daraufhin selbständig nachgeladen.

Eigene Befehle: Mit dem Programm »MG SPEICHER« oder mit einem Monitor auf Diskette gespeicherte Programme müssen mit einem bestimmten Namen versehen sein (siehe oben). Mit dem Menüpunkt »Parameter eingeben« des Hauptprogrammes müssen Sie die zu jedem Modul folgenden Angaben machen: Die Anzahl der im Modul enthaltenen Befehle, den Namen, mit dem der Befehl aufgerufen wird, zu jedem Namen die Startadresse, die Endadresse und die Anzahl der Datenbytes. Die Parameter werden dann in einem File mit dem Namen »xMODUL« gespeichert. X ist dabei wieder die Nummer des Moduls.

Erweiterungen erstellen: Geben Sie einfach die Nummern der gewünschten Module nacheinander ein. Die Eingabe von »F« storniert das jeweils letzte Modul. Sind alle Erweiterungen im Speicher, beenden Sie die Eingaben mit »E«. Als nächsten Schritt geben Sie die Startadresse der Tabelle sowie der Befehle ein. Wenn die Befehle direkt nach der Tabelle liegen sollen, dann bestätigen Sie bei »Befehle« die vom Computer vorgeschlagene Adresse mit der RETURN-Taste. Daraufhin wird die Erweiterung generiert, jedoch noch nicht gespeichert. Sie steht jetzt im Speicher des Computers an den von Ihnen angegebenen Adressen.

Speichern einer Erweiterung: Mit der Taste F3 gelangen Sie in das Speicher-Menü. Zwei Angaben werden hier von Ihnen erwartet: Der Name, unter dem die Befehlstabelle und die Interpreter-Erweiterung gespeichert wird sowie der Name, unter dem die Befehle gespeichert werden. Wenn Tabelle und Befehle direkt hintereinander liegen, genügt zum Speichern natürlich ein Name. In diesem Fall geben Sie bei »Befehle« nichts ein, sondern tippen einfach nur RETURN. Daraufhin wird die Erweiterung auf Diskette gespeichert.

Protokoll: Alle wichtigen Daten der im Speicher des C 64 liegenden Erweiterung werden noch einmal gezeigt. Eventuell sollten Sie hier noch einmal prüfen, ob es zwischen der Tabelle und den Befehlen keine Überschneidung gegeben hat. Das Protokoll können Sie auf Drucker ausgegeben.

Starten einer Erweiterung: Bringen Sie Ihren C 64 in den Einschaltzustand und laden Sie Ihre Erweiterung mit LOAD "name",8,1. Danach geben Sie NEW ein, um alle Zeiger wieder richtig zu setzen. Ist die Tabelle von den Befehlen getrennt, so laden Sie diese nun nach und geben auch danach wieder ein NEW ein. Der Start der Erweiterung, also das Einbinden in die Interpreterschleife, geschieht durch den Befehl SYS (Startadresse der Tabelle). Nach dem READY des Computers sind die neuen Befehle aktiv.

Hinweise für Programmierer: Wenn Sie selbst in Maschinensprache programmieren und sich einige Module selbst anfertigen wollen, beachten Sie bitte folgende Punkte: Ein Modul muß, bevor es mit Namen aufgerufen wird, auch über einen SYS-Befehl zu erreichen sein. Ein entsprechender Aufruf wäre zum Beispiel: `SYS (Adresse) P1,P2,P3.`

Vor dem ersten Parameter darf also kein Komma stehen, denn dieser müßte ja sonst nach dem Befehlswort mit eingegeben werden. Achten Sie bitte darauf, daß ein Modul mit mehreren Befehlen in sich geschlossen lauffähig sein muß, also nicht in ein anderes Modul verzweigt.

Module

Die in dieser Ausgabe veröffentlichten Module (Listings 5 bis 13) sind Beispiele, die die Fähigkeiten des Hypra-Basic demonstrieren und den Umgang veranschaulichen sollen. Die Erklärungen und Parameter dazu sind jeweils beim Modul selbst in der Bildunterschrift enthalten.

Bestimmt werden auch Sie Module für Hypra-Basic entwickeln. Die besten Module, die uns zugeschickt werden, werden wir gegen ein entsprechendes Honorar abdrucken. Somit kann sich jeder in kurzer Zeit eine Bibliothek von Basic-Befehlen, Tools und sonstigen nützlichen Routinen anlegen. Erstrebenswert wäre ein Standard-Basic für den C 64, für das dann auch Listings veröffentlicht würden. In diesem Sinne wünschen wir Ihnen viel Erfolg mit Hypra-Basic.

(R. Aretz/og)

```
programm : basic .ass        3138 3383
-------------------------------------

3138 : 20 fd ae 20 8a ad 20 f7  8d
3140 : b7 a0 00 84 f7 84 f8 b1  c3
3148 : 14 85 02 b9 88 31 c5 02  04
3150 : f0 0a c8 c0 95 d0 f4 a9  96
3158 : 00 85 f7 60 84 a7 c0 1d  e7
3160 : b0 07 a9 01 85 f7 4c 87  77
3168 : 31 c0 65 b0 07 a9 02 85  3a
3170 : f7 4c 87 31 a9 03 85 f7  4e
3178 : a9 01 85 f8 a0 01 b1 14  23
3180 : 85 f9 c8 b1 14 85 fa 60  84
3188 : 0a 4a 2a 6a 8a aa 98 a8  41
3190 : ba 9a 08 28 48 68 00 40  e7
3198 : 60 ea 18 38 58 78 b8 d8  58
31a0 : f8 88 c8 ca e8 69 29 c9  7a
31a8 : e0 c0 49 a9 a2 a0 09 e9  97
31b0 : 65 25 06 24 c5 e4 c4 c6  d2
31b8 : 45 e6 a5 a6 a4 46 05 26  8b
31c0 : 66 e5 85 86 84 75 35 16  40
31c8 : d5 d6 55 f6 b5 b4 56 15  ç1
31d0 : 36 76 f5 95 94 61 21 c1  ce
31d8 : 41 a1 01 e1 81 71 31 d1  72
31e0 : 51 b1 11 f1 91 10 30 50  87
31e8 : 70 90 b0 d0 f0 6d 2d 0e  32
31f0 : 2c cd ec cc ce 4d ee ad  46
31f8 : ae ac 4e 0d 2e 6e ed 8d  5b
3200 : 8e 8c 7d 3d 1e dd de 5d  e2
3208 : fe bd bc 5e 1d 3e 7e fd  99
3210 : 9d 79 39 d9 59 b9 be 19  84
3218 : f9 99 20 4c 6c 20 8a ad  bd
3220 : 20 f7 b7 a5 14 85 f7 a5  77
3228 : 15 85 f8 20 fd ae 20 8a  2d
3230 : ad 20 f7 b7 a5 14 85 f9  e7
3238 : a5 15 85 fa 20 fd ae 20  15
3240 : 8a ad 20 f7 b7 a5 14 85  ac
3248 : 9b a5 15 85 9c a0 00 a5  c6
3250 : 01 85 02 78 a9 35 85 01  00
3258 : b1 f7 91 9b e6 f7 d0 02  52
3260 : e6 f8 e6 9b d0 02 e6 9c  e1
3268 : a5 f8 c5 fa d0 ea a5 f7  45
3270 : c5 f9 d0 e4 a5 02 85 01  85
3278 : 58 20 79 00 a8 d0 9e 60  8b
3280 : 20 9e ad 20 82 b7 a6 22  23
3288 : a4 23 20 bd ff 20 fd ae  d4
3290 : 20 9e b7 a0 01 20 ba ff  fd
3298 : a9 00 20 d5 ff 86 f7 84  21
32a0 : f8 60 20 9e ad 20 82 b7  fa
32a8 : a6 22 a4 23 20 bd ff 20  1d
32b0 : fd ae 20 9e b7 20 ba ff  48
32b8 : 20 fd ae 20 8a ad 20 f7  0d
32c0 : b7 a5 14 85 f7 a5 15 85  0c
32c8 : f8 20 fd ae 20 8a ad 20  73
32d0 : f7 b7 a6 14 e8 d0 02 e6  ba
32d8 : 15 a4 15 a9 f7 20 d8 ff  9e
32e0 : 60 a9 ec 8d 08 03 a9 32  a5
32e8 : 8d 09 03 60 20 73 00 c9  f8
32f0 : 7f 90 06 20 79 00 4c e7  d6
32f8 : a7 a9 82 85 f7 a9 33 85  6a
3300 : f8 a0 00 b1 f7 85 f9 18  42
3308 : 69 05 85 fa e6 f7 d0 02  2a
3310 : e6 f8 b1 f7 f0 dd d1 7a  18
3318 : d0 07 c8 c4 f9 f0 13 d0  4b
3320 : f1 18 a5 f7 65 fa 85 f7  ba
3328 : a5 f8 69 00 85 f8 a0 00  46
3330 : f0 cf b1 f7 85 55 c8 b1  fd
3338 : f7 85 56 c8 18 a5 7a 65  04
3340 : f9 85 7a 90 02 e6 7b a9  45
3348 : 82 85 57 a9 33 85 58 a5  a4
3350 : 01 85 02 78 a9 35 85 01  00
3358 : b1 f7 85 59 c8 b1 f7 85  96
3360 : 5a c8 b1 f7 aa a0 00 b1  9d
3368 : 59 91 57 c8 d0 f9 e6 5a  a6
3370 : e6 58 ca 10 f0 a5 02 85  86
3378 : 01 58 20 54 00 4c ae a7  a4
3380 : 00 00 00 00 00 02 da 00  fc
```

Listing 1. »BASIC .ASS« geben Sie bitte mit dem MSE ein

```
5 IF FG<>0 THEN 10                                  <156>
7 FG=1:LOAD "BASIC .ASS",8,1                        <164>
10 DIM A%(100):AN=1:MD=1:POKE 53280,0:POKE
   53281,0:F=1:POKE 646,F:MP=12600                  <229>
15 POKE 55,56:POKE 56,49                            <218>
20 HL$="{39SPACE}"                                  <200>
30 H$="************************************
   ****"                                            <241>
40 REM **** HAUPTMENUE *******                      <235>
50 PRINT CHR$(147)TAB(9)"{RVSON,SPACE}HAUP
   TMENUE{SPACE,RVOFF,SPACE}HYPRA-BASIC":P
   RINT H$:AN=1:MD=1                                <121>
60 PRINT"FOLGENDE OPTIONEN SIND MOEGLICH:"
   :PRINT H$:PRINT                                  <189>
70 PRINT TAB(5)"{RVSON}[F1]{RVOFF,SPACE}ER
   WEITERUNG ERSTELLEN                              <134>
80 PRINT TAB(5)"{DOWN,RVSON}[F3]{RVOFF,SPA
   CE}ERWEITERUNG ABSPEICHERN                       <209>
90 PRINT TAB(5)"{DOWN,RVSON}[F5]{RVOFF,SPA
   CE}PARAMETER EINGEBEN                            <171>
100 PRINT TAB(5)"{DOWN,RVSON}[F7]{RVOFF,SP
    ACE}PARAMETER ANZEIGEN"                         <089>
110 PRINT TAB(5)"{DOWN,RVSON}[F2]{RVOFF,SP
    ACE}PROTOKOLL":PRINT:PRINT H$                   <167>
115 PRINT TAB(10)"BITTE WAEHLEN SIE !!"             <108>
120 POKE 198,0:WAIT 198,1:GET S$:S=ASC(S$)          <176>
130 IF S<133 OR S>137 THEN 120                      <225>
140 S=S-132                                         <047>
150 ON S GOSUB 210,410,600,800,11000                <054>
160 GOTO 50                                         <114>
200 REM ***** ERWEITERUNG ERSTELLEN ***             <063>
210 GOSUB 1010                                      <198>
220 POKE 211,8:POKE 214,0:SYS 58640:PRINT"
    SIND SIE SICHER ? (J/N)"                        <020>
230 POKE 198,0:WAIT 198,1:GET S$:IF S$<>"J
    "OR AN=0 THEN RETURN                            <154>
240 GOSUB 10010                                     <028>
250 PRINT CHR$(147)"SIE MUESSEN JETZT ZWEI
     SPEICHERBEREICHE"                              <246>
260 PRINT"AUSWAEHLENEN":PRINT:PRINT"1) BER
    EICH FUER BEFEHLSTABELLE.DIESER"                <010>
270 PRINT TAB(3)"DARF NUR IM 'NORMALEN RAM
    ' LIEGEN.                                       <170>
280 PRINT:PRINT"2) BEREICH FUER DIE ERWEIT
    ERUNG.HIER                                      <044>
290 PRINT TAB(3)"IST DER SPEICHERBEREICH B
    ELIEBIG.":PRINT                                 <248>
292 PRINT"FALLS BEIDE PROGRAMMTEILE NACHEI
    NANDER"                                         <170>
293 PRINT"STEHEN SOLLEN,BITTE BEI 'BEFEHLE
    ' NUR"                                          <194>
294 PRINT"<RETURN> DRUECKEN.":PRINT H$              <063>
295 PRINT"DIE BEFEHLSTABELLE BRAUCHT";HG"
    BYTES":PRINT H$                                 <094>
300 INPUT"1) TABELLE:";SD$:SD=VAL(SD$):IF
    SD=0 THEN 250                                   <199>
305 PRINT TAB(12)SD+HG                              <246>
310 INPUT"{UP}2) BEFEHLE:";MA$:MA=VAL(MA$)
    :MS=MA                                          <050>
320 IF MA=0 THEN RETURN                             <177>
330 PRINT CHR$(147)"ERWEITERUNG WIRD GENER
    IERT":PRINT                                     <077>
340 LA=SD+162+HA:GOSUB 10410                        <051>
350 FOR KI=1 TO AN:MO%=A%(KI):PRINT MO%,            <117>
360 HM=INT(MO%/70)+1:IF MD<>HM THEN GOSUB
    10210                                           <037>
370 GOSUB 10660:NEXT:RETURN                         <060>
400 REM ***** PROGRAMM ABSPEICHERN **               <114>
410 PRINT CHR$(147)TAB(10)"ERWEITERUNG SPE
    ICHERN":PRINT H$                                <081>
420 PRINT"BEIM ABSPEICHERN KOENNEN ZWEI VE
    RSCHIE- DENE ";                                 <026>
430 PRINT"FILES ABGELEGT WERDEN.":PRINT             <141>
440 PRINT"1) BEFEHLSTABELLE"                        <120>
450 PRINT"2) MODULE":PRINT                          <149>
460 PRINT"DIES IST ERFORDERLICH,WENN BEIDE
     PRO-"                                          <038>
470 PRINT"GRAMMTEILE UNABHAENGIG VONEINAND
    ER                                              <144>
480 PRINT"EXISTIEREN.FALLS SIE BEIDE PROGR
    AMME"                                           <097>
490 PRINT"GEMEINSAM GENERIERT HABEN,GEBEN
    SIE UN-"                                        <238>
500 PRINT"TER MODULE EINEN LEERSTRING EIN.
    ":PRINT H$                                      <109>
510 INPUT"TABELLE:";NA$                             <112>
520 INPUT"BEFEHLE:";NB$:PRINT H$                    <183>
525 PRINT"SIND SIE SICHER,DASS DIE RICHTIG
    E DIS- "                                        <197>
527 PRINT"KETTE IM LAUFWERK IST? (J/N)"             <219>
528 POKE 198,0:WAIT 198,1:GET A$:IF A$="N"
    THEN RETURN                                     <070>
530 IF NB$=""THEN SYS(MP+362)"@:"+NA$,8,SD
    ,MA+1:RETURN                                    <187>
540 SYS(MP+362)"@:"+NA$,8,SD,BT+1                   <155>
550 SYS(MP+229)MS,MA+1,MP+2000                      <153>
560 CLOSE 1:OPEN 1,8,1,"@:"+NB$:PRINT#1,CH
    R$(MS-INT(MS/256)*256);CHR$(MS/256);            <088>
570 N=MA+1-MS                                       <220>
580 FOR I=0 TO N-1:PRINT#1,CHR$(PEEK(MP+20
    00+I));:NEXT:CLOSE 1                            <070>
590 RETURN                                          <140>
600 REM ******* PARAMETER EINGEBEN ***              <218>
610 PRINT CHR$(147)TAB(10)"PARAMETER EINGE
    BEN":PRINT H$                                   <058>
615 INPUT"MODULNUMMER{5SPACE}:";MO$                 <103>
620 INPUT"WIEVIELE BEFEHLE:";AB                     <158>
630 PRINT H$                                        <100>
640 FOR I=1 TO AB                                   <048>
650 POKE 211,0:POKE 214,5:SYS 58640:PRINT
    I"{LEFT}) ";:INPUT "NAME:";NA$(I):PRIN
    T H$                                            <087>
660 INPUT"{4SPACE}STARTADRESSE:";AS$(I)             <168>
670 INPUT"{4SPACE}ENDADRESSE{2SPACE}:";AE$
    (I)                                             <146>
680 INPUT"{4SPACE}DATENBYTES{2SPACE}:";AD$
    (I)                                             <016>
690 PRINT H$:NEXT                                   <135>
700 PRINT TAB(10)"SIND SIE SICHER (J/N)"            <228>
710 POKE 198,0:WAIT 198,1:GET A$:IF A$="N"
    THEN RETURN                                     <252>
720 CLOSE 1:OPEN 1,8,1,"@:"+MO$+"MODUL,S"           <216>
730 PRINT#1,STR$(AB)                                <097>
740 FOR I=1 TO AB:PRINT#1,NA$(I):PRINT#1,A
    S$(I):PRINT#1,AE$(I):PRINT#1,AD$(I)             <206>
750 NEXT:CLOSE 1:RETURN                             <210>
800 REM ****** PARAMETER ANZEIGEN ****              <005>
810 PRINT CHR$(147)TAB(10)"PARAMETER ANZEI
```

Listing 2. »HYPRA-BASIC«, das Hauptprogramm

```
     GEN":PRINT H$                          <135>
820 INPUT"VON MODULNUMMER:";M1              <135>
830 INPUT"BIS MODULNUMMER:";M2:PRINT H$     <231>
840 FOR K=M1 TO M2:POKE 214,5:SYS 58640     <027>
845 FOR I=1 TO 15:PRINT HL$:NEXT            <029>
850 POKE 211,0:POKE 214,5:SYS 58640         <113>
860 MO%=K:GOSUB 10530:PRINT"MODULNUMMER:"K
    :PRINT"ANZAHL DER BEFEHLE:"AB:PRINT H$ <085>
870 FOR I=1 TO AB                           <024>
880 POKE 214,9:SYS 58640:PRINT I"{LEFT}) N
    AME:"NA$(I);LEFT$(HL$,10):PRINT         <122>
890 PRINT"STARTADRESSE:"AS(I);LEFT$(HL$,10
    )                                       <122>
900 PRINT"ENDADRESSE{2SPACE}:"AE(I);LEFT$(
    HL$,10)                                 <010>
910 PRINT"DATENBYTES{2SPACE}:"AD(I);LEFT$(
    HL$,10)                                 <138>
930 PRINT H$:PRINT TAB(6)"WEITER MIT BELIE
    BIGER TASTE"                            <224>
940 POKE 198,0:WAIT 198,1:NEXT:NEXT:RETURN <108>
999 END                                     <239>
1000 REM ***** MODULE AUSWAEHLEN *****      <221>
1010 PRINT CHR$(147)TAB(10)"MODULE AUSWAEH
     LEN!"                                  <187>
1020 PRINT H$                               <236>
1040 PRINT"BITTE MODULNUMMERN EINGEBEN:"    <190>
1050 PRINT H$                               <012>
1060 FOR I=1 TO 19:PRINT HL$:NEXT           <250>
1070 PRINT H$:PRINT TAB(5)"{RVSON,SPACE}E{
     SPACE,RVOFF,SPACE}= ENDE{9SPACE,RVSON
     ,SPACE}F{SPACE,RVOFF,SPACE}= FEHLER"; <222>
1080 POKE 211,30:POKE 214,2:SYS 58640:POKE
      19,1:INPUT MN$:POKE 19,0:PRINT        <029>
1090 IF LEFT$(MN$,1)="E"THEN AN=AN-1:RETUR
     N                                      <123>
1100 IF LEFT$(MN$,1)="F"AND AN>1 THEN AN=A
     N-1:MN$="{3SPACE}":GOTO 1130           <137>
1110 IF LEN(MN$)>3 OR VAL(MN$)=0 THEN 1080 <235>
1120 A%(AN)=VAL(MN$)                        <096>
1130 Z=INT(AN/10)+4:S=(AN-(Z-4)*10)*4:POKE
      211,S:POKE 214,Z:SYS 58640:PRINT MN$ <046>
1150 IF MN$<>"{3SPACE}"THEN AN=AN+1         <021>
1160 GOTO 1080                              <085>
10000 REM **** LAENGE DECODE BESTIMMEN *    <021>
10010 HG=0:HA=0:AB=0:ZB=0:FOR JI=1 TO AN    <204>
10015 HM=INT(A%(JI)/70)+1:IF HM<>MD THEN G
      OSUB 10210                            <228>
10020 CLOSE 1:OPEN 1,8,0,MID$(STR$(A%(JI))
      ,2,4)+"MODUL,S"                       <054>
10030 INPUT#1,A$:H=VAL(A$):ZB=ZB+H          <048>
10040 FOR J=1 TO H                          <124>
10050 INPUT#1,H1$,H2$(J),H3$(J),H4$(J):HA=
      HA+LEN(H1$)+6                         <236>
10060 NEXT                                  <164>
10070 HB=(VAL(H3$(H))+VAL(H4$(H)))-VAL(H2$
      (1)):IF HB>HG THEN HG=HB              <110>
10090 CLOSE 1:NEXT                          <182>
10100 HG=INT((HG/256)+1)*256:HG=HG+HA+170:
      RETURN                                <249>
10200 REM **** MODULDISK AUSGEBEN ****      <061>
10210 H1=PEEK(211):H2=PEEK(214)             <226>
10212 FOR X=1984 TO 2023:POKE X-1156,PEEK(
      X):POKE X,32:X1=X1+1:NEXT             <105>
10215 POKE 211,0:POKE 214,24:SYS 58640      <169>
10220 PRINT TAB(3)"{GREY 2}BITTE MODULDISK
       NR."HM" EINLEGEN! ";                 <206>
10230 MD=HM:POKE 646,F:POKE 198,0:WAIT 198
      ,1                                    <242>
10240 FOR X=1984 TO 2023:POKE X,PEEK(X-115
      6):POKE X+54272,F:NEXT:POKE 211,H1    <141>
10250 POKE 214,H2:SYS 58640:RETURN          <159>
10300 REM ****** PROGRAMM LADEN *****       <208>
10310 HI=INT(LA/256):LO=LA-256*HI           <019>
10330 SYS(57812)NA$,8,0:POKE 780,0:POKE 78
      1,LO:POKE 782,HI:SYS 65493            <071>
10340 RETURN                                <238>
10400 REM ****** DECODE ROUTINE HOLEN **    <183>
10410 H=LA:LA=SD:SYS(MP+229)13025,13184,SD <203>
10430 LA=H:H=SD+11:HI=INT(H/256):LO=H-256*
      HI:POKE SD+1,LO:POKE SD+6,HI          <104>
10440 H=SD+161:HI=INT(H/256):LO=H-256*HI:P
      OKE SD+25,LO:POKE SD+29,HI            <233>
10460 HI=INT(LA/256):LO=LA-256*HI:POKE SD+
      103,LO:POKE SD+107,HI                 <246>
10470 POKE LA-1,0:BT=SD+161:RETURN          <069>
10500 REM ***** PARAMETER LADEN *****       <182>
10530 FL$=MID$(STR$(MO%)+"MODUL,S,R",2,13) <058>
10540 CLOSE 2:OPEN 2,8,2,FL$                <120>
10550 INPUT#2,A$:AB=VAL(A$)                 <063>
10560 FOR JI=1 TO AB:INPUT#2,NA$(JI):INPUT
      #2,A$:AS(JI)=VAL(A$)                  <174>
10570 INPUT#2,A$:AE(JI)=VAL(A$)             <214>
10580 INPUT#2,A$:AD(JI)=VAL(A$)             <222>
10590 NEXT                                  <186>
10600 CLOSE 2:RETURN                        <065>
10610 REM **** BEFEHLSTABELLE BEARBEITEN** <208>
10615 YY=LEN(NA$(I)):POKE BT,YY             <072>
10620 FOR K=1 TO YY:POKE BT+K,ASC(MID$(NA$
      (I),K,1)):NEXT:BT=BT+K                <247>
10630 HI=INT(PZ/256):LO=PZ-256*HI:POKE BT,
      LO:POKE BT+1,HI                       <100>
10635 HI=INT(MA/256):LO=MA-256*HI:POKE BT+
      2,LO:POKE BT+3,HI                     <244>
10636 LE=AE(AB)+AD(AB)-AS(1):POKE BT+4,INT
      (LE/256):BT=BT+5                      <070>
10640 RETURN                                <028>
10650 REM ********** VERSCHIEB **********   <218>
10660 GOSUB 10530:A$=STR$(MO%)+"ERW .ASS":
      NA$=MID$(A$,2,13):GOSUB 10310:PZ=LA   <239>
10670 N1=PZ:FOR I=1 TO AB:GOSUB 10610:N=AS
      (I)                                   <216>
10680 SYS MP,PZ:A=PEEK(247)                 <238>
10690 IF A<>3 THEN 10740                    <109>
10700 A1=PEEK(249)+256*PEEK(250)            <072>
10710 IF A1<AS(1)OR A1>AE(AB)+AD(AB)THEN 1
      0750                                  <184>
10720 A1=A1-AS(1):A1=N1+A1:HI=INT(A1/256):
      LO=A1-256*HI                          <145>
10730 POKE PZ+1,LO:POKE PZ+2,HI             <250>
10740 IF A=0 THEN A=1                       <076>
10750 PZ=PZ+A:N=N+A:IF N<>AE(I)+1 THEN 106
      80                                    <009>
10760 PZ=PZ+AD(I):NEXT                      <187>
10770 SYS(MP+229)LA,PZ,MA                   <017>
10780 MA=MA+LE+1:RETURN                     <165>
11000 REM ****** PROTOKOLL ********         <118>
11010 PRINT CHR$(147)TAB(12)"PROTOKOLL":PR
      INT H$                                <002>
11020 GN=3                                  <130>
11025 CLOSE 1:OPEN 1,GN                     <077>
11030 PRINT#1,"ERWEITERUNG HYPRA-BASIC VER
      SION 1.1":PRINT#1,H$                  <004>
11040 PRINT#1,"ANZAHL DER BEFEHLE:{9SPACE}
      ";ZB                                  <112>
11050 PRINT#1,"STARTADRESSE BEFEHLSTABELLE
      :";SD                                 <119>
11060 PRINT#1,"STARTADRESSE DER MODULE:{4S
      PACE}";MS:PRINT#1,H$                  <012>
11070 PRINT#1,"BYTES FUER TABELLE:{9SPACE}
      ";HG                                  <008>
11080 PRINT#1,"BYTES FUER BEFEHLE:{9SPACE}
      ";MA-MS:PRINT#1,H$                    <039>
11082 PRINT#1,"DIE ERWEITERUNG ENTHAELT FO
      LGENDE MODULE":PRINT#1                <036>
11085 FOR I=1 TO KI-1:PRINT#1,A%(I);:NEXT:
      PRINT#1                               <053>
11090 IF FK=1 THEN FK=0:GN=3:CLOSE 1:RETUR
      N                                     <245>
11100 PRINT#1,H$:PRINT#1,"{2SPACE}AUSGABE
      AUF DRUCKER (J/N)"                    <102>
11110 POKE 198,0:WAIT 198,1:GET A$:IF A$="
      N"THEN CLOSE 1:RETURN                 <011>
11120 GN=4:FK=1:CLOSE 1:GOTO 11025          <181>
```

© 64'er

Listing 2. »HYPRA-BASIC«, das Hauptprogramm (Schluß)

```
10 REM ****** DOKUMENT ******                  <240>
20 POKE 53280,0:POKE 53281,0:DIM AN$(16),A
   S$(16),AE$(16),AD$(16)                      <112>
30 INPUT"{CLR,WHITE}GERAETENUMMER 3=BILD,4
   =DRUCKER ";G                                <035>
40 INPUT"{DOWN}STARTMODUL:";SM                 <149>
50 INPUT"{DOWN}ENDMODUL:";EM:PRINT"{CLR}";    <233>
60 FOR I=SM TO EM:GOSUB 500                    <094>
70 OPEN 1,G:I$=STR$(I)+"){2SPACE}"             <231>
80 PRINT#1,LEFT$(I$,4);:A$=" "                 <051>
100 FOR K=1 TO AB:IF K<>1 THEN A$="{5SPACE
    }"                                         <050>
110 PRINT#1,A$;AN$(K)"{8SPACE}"AS$(K)" - "
    AE$(K)"{5SPACE}"AD$(K)                     <227>
120 NEXT                                       <130>
130 PRINT#1,"*******************************
    **********":CLOSE 1                        <164>
135 IF G=3 THEN PRINT"{DOWN,9SPACE}WEITER
    MIT TASTE !!":POKE 198,0:WAIT 198,1:PR
    INT                                        <213>
140 NEXT                                       <150>
150 IF G=3 THEN 60                             <195>
160 END                                        <162>
500 REM **** SUBR. PARAMETER HOLEN ****        <205>
505 NA$=MID$(STR$(I),2,4)+"MODUL,S,R"          <232>
510 OPEN 1,8,3,NA$                             <098>
520 INPUT#1,AB$:AB=VAL(AB$)                    <213>
530 FOR K=1 TO AB                              <210>
540 INPUT#1,AN$(K),AS$(K),AE$(K),AD$(K):NE
    XT:CLOSE 1:RETURN                          <004>
```

© 64'er

Listing 3. »DOKUMENT« zeigt die Module einer Diskette an

```
10 POKE 53280,0:POKE 53281,0:PRINT"{CLR,WH
   ITE}"                                       <192>
20 INPUT "MODULNUMMER:";A$                     <069>
30 INPUT "{DOWN}STARTADRESSE:";AS              <120>
35 PRINT"{DOWN}ENDADRESSE DES LETZTEN BEFE
   HLS + DATEN-":PRINT"BYTES DES LETZTEN B
   EFEHLS!                                     <016>
40 INPUT"{DOWN}";AE:AE=AE+1                    <036>
45 INPUT"{DOWN,3SPACE}SIND SIE SICHER (J/N
   ){2SPACE}E=ENDE ";S$:IF S$="J"THEN 50       <241>
47 IF S$="E"THEN END                           <044>
48 GOTO 10                                     <226>
50 H1=INT(AS/256):L1=AS-256*H1                 <095>
60 H2=INT(AE/256):L2=AE-256*H2                 <248>
```

►

```
70 SYS(57812)"@:"+A$+"ERW .ASS",8:POKE 193
   ,L1:POKE 194,H1                          <219>
80 POKE 174,L2:POKE 175,H2:SYS 62957        <057>

© 64'er
```

Listing 4. »MG SPEICHER«. Mit ihm kann man Maschinensprache-Programme auf Diskette bringen. Es geht aber auch mit einem Monitor oder dem MSE.

```
programm : 1erw .ass          c000 c05b
---------------------------------------

c000 : a5 39 8d 4c c0 a5 3a 8d   6c
c008 : 4d c0 a5 7a 85 f7 a5 7b   14
c010 : 85 f8 a5 f7 85 7a a5 f8   2e
c018 : 85 7b a9 00 85 c6 20 87   e3
c020 : ea a5 c6 f0 f5 ad 77 02   5b
c028 : 85 02 4c 35 c0 20 79 00   5b
c030 : f0 e0 20 fd ae 20 8a ad   ca
c038 : 20 f7 b7 20 fd ae 20 9e   59
c040 : b7 e4 02 d0 e8 a9 00 85   eb
c048 : c6 4c a3 a8 00 00 ad 4c   82
c050 : c0 85 14 ad 4d c0 85 15   a9
c058 : 4c a3 a8 ff 00 ff 00 fd   9c
```

Listing 5. Modulnummer 1:
Anzahl der Befehle: 2

1) KEY 49152-49227 2
Funktion: Sprung in angegebene Zeilennummer nach Tastendruck mit nachfolgendem ASCII-Code.
Syntax: KEY zeinr,ascii(,zeinr,ascii...)
Anwendung: Bei Menüsteuerungen, keine Fehleingabe möglich, da nur Tasten, die im Befehlaufruf stehen, beachtet werden; alle anderen werden ignoriert.
Beispiel: KEY 12000,65,13000,66,14000,67
In diesem Beispiel wird beim Drücken der Taste 65 (A) in die Zeile 12000 verzweigt, bei 66 (B) nach 13000 etc.

2) RETKEY 49230-49242 0
Funktion: Erreicht das Programm diesen Befehl, wird in die Key-Zeile zurückgesprungen ähnlich einem RETURN.
Syntax: RETKEY   keine Parameter


```
programm : 2erw .ass          c000 c0c3
---------------------------------------

c000 : 20 9e b7 e0 19 30 03 4c   31
c008 : 85 c0 8e 3c 03 86 f7 20   9d
c010 : fd ae 20 9e b7 e0 28 10   83
c018 : 6c 8e 3d 03 20 fd ae 20   68
c020 : 9e b7 e0 19 10 5f e8 8e   b2
c028 : 3e 03 20 fd ae 20 9e b7   85
c030 : e0 28 10 51 e8 8e 3f 03   59
c038 : ad 3c 03 cd 3e 03 30 02   3f
c040 : 10 43 ad 3d 03 cd 3f 03   a6
c048 : 30 02 10 39 ad 3d 03 85   80
c050 : d3 ad 3c 03 85 d6 20 10   19
c058 : e5 ac 3d 03 b1 d1 10 04   35
c060 : 29 7f 10 02 09 80 91 d1   0c
c068 : c8 cc 3f 03 d0 ee 18 a5   f7
c070 : d1 69 28 90 02 e6 d2 85   c0
c078 : d1 ee 3c 03 ad 3c 03 cd   94
c080 : 3e 03 d0 d5 60 a2 0e 4c   1b
c088 : 37 a4 20 00 c0 a9 00 85   7e
c090 : c6 85 a7 20 fd ae 20 9e   1a
c098 : b7 86 f8 a6 f8 a5 f7 8d   5d
c0a0 : 3c 03 a0 00 ea ea c8 d0   51
c0a8 : fb ca d0 f6 20 4c c0 e6   50
c0b0 : a7 a5 c6 f0 e6 a5 a7 4a   c8
c0b8 : b0 08 a5 f7 8d 3c 03 20   dc
c0c0 : 4c c0 60 00 ff 00 ff 00   84
```

Listing 6. Modulnummer: 2
Anzahl der Befehle: 2

1) INVERS 49152-49289 0
Funktion: Invertieren eines rechteckigen Bildschirm-Ausschnitts.
Syntax: INVERS AZ,AS,EZ,ES
Beispiel: INVERS 2,3,10,25
Der Bildschirm-Bereich Zeile 2, Spalte 3 (obere linke Ecke) bis Zeile 10, Spalte 25 (untere Ecke) werden invertiert.

2) BLINK 49290-49346 0
Funktion: Rechteckiger Bildschirm-Ausschnitt blinkt mit angegebener Geschwindigkeit.
Syntax: BLINK AZ,AS,EZ,ES,SPEED
Beispiel: BLINK 2,3,10,25,100
Der Bildschirmbereich wie unter (1) blinkt mit der Geschwindigkeit 100 (1-255), je größer die Zahl SPEED, desto langsamer blinkt der Bildschirm.

```
programm : 3erw .ass          c000 c16b
---------------------------------------

c000 : 20 9e b7 8e 3c 03 20 fd   87
c008 : ae 20 9e b7 8e 3d 03 20   84
c010 : fd ae 20 9e b7 8e 3e 03   2f
c018 : 20 fd ae 20 9e b7 86 f7   98
c020 : ad 3c 03 85 d6 ad 3d 03   33
c028 : 85 d3 20 10 e5 20 24 ea   67
c030 : ac 3d 03 a2 40 b1 d1 c9   fc
c038 : 5d d0 02 a2 5b 8a 91 d1   c6
c040 : a5 f7 91 f3 c8 a2 40 ce   04
c048 : 3e 03 d0 e9 60 20 9e b7   6a
c050 : 8e 3c 03 20 fd ae 20 9e   d4
c058 : b7 8e 3d 03 20 fd ae 20   f3
c060 : 9e b7 8e 3e 03 20 fd ae   cc
c068 : 20 9e b7 86 f7 ad 3c 03   7a
c070 : 85 d6 ad 3d 03 85 d3 20   5f
c078 : 10 e5 20 24 ea a2 5d 18   71
c080 : ad 3c 03 6d 3e 03 c9 1a   11
c088 : 10 29 ac 3d 03 b1 d1 c9   98
c090 : 40 d0 02 a2 5b 8a 91 d1   01
c098 : a5 f7 91 f3 18 a5 d1 69   e4
c0a0 : 28 90 04 e6 d2 e6 f4 85   31
c0a8 : d1 85 f3 a2 5d ce 3e 03   d8
c0b0 : d0 d8 60 a2 0e 4c 37 a4   c2
c0b8 : 20 9e b7 8e 3f 03 20 fd   70
c0c0 : ae 20 9e b7 8e 40 03 20   54
c0c8 : fd ae 20 9e b7 8e 41 03   f3
c0d0 : 20 fd ae 20 9e b7 8e 42   05
c0d8 : 03 20 fd ae 20 9e b7 86   23
c0e0 : f7 ad 3f 03 8d 3c 03 ad   00
c0e8 : 40 03 8d 3d 03 ad 41 03   5e
c0f0 : 8d 3e 03 20 20 c0 ad 42   a4
c0f8 : 03 8d 3e 03 20 6d c0 18   52
c100 : ad 3f 03 6d 42 03 aa ca   38
c108 : 8e 3c 03 ad 41 03 8d 3e   0a
c110 : 03 20 20 c0 ad 3f 03 8d   3f
c118 : 3c 03 18 ad 41 03 6d 40   f4
c120 : 03 aa ca 8e 3d 03 ad 42   24
c128 : 03 8d 3e 03 20 6d c0 ad   ae
c130 : 3f 03 85 d6 ad 40 03 85   21
c138 : d3 20 10 e5 a4 d3 a9 70   4c
c140 : 91 d1 98 18 6d 41 03 48   60
c148 : a8 88 a9 6e 91 d1 18 ad   d0
c150 : 3f 03 6d 42 03 aa ca 86   72
c158 : d6 20 10 e5 ac 40 03 a9   2b
c160 : 6d 91 d1 68 a8 88 a9 7d   88
c168 : 91 d1 60 ff 00 ff 00 ff   fa
```

Listing 7. Modulnummer: 3
Anzahl der Befehle: 3

1) WLINE 49152-49228 0
Funktion: Zeichnet eine waagrechte Linie in einen LOW-RES-Bildschirm. Eventuelle Kreuzungen mit senkrechten Linien werden berücksichtigt.
Syntax: WLINE AZ,AS,Länge,Farbe
Beispiel: WLINE 2,4,10,1
Mit diesem Beispiel wird eine waagrechte Linie, die im Punkt (2,4) beginnt und eine Länge von 10 besitzt, mit der Farbe 1 (weiß) gezeichnet. Dieser wie auch die folgenden beiden Befehle beziehen sich auf einen LOW-Resolution-Bildschirm. Sie eignen sich besonders für eine Menüaufbereitung oder zur Anzeige von Tabellen.

2) SLINE 49229-49335 0
Funktion: Zeichnet senkrechte Linie in LOW-RES-Bildschirm.
Syntax: SLINE AZ,AS,Höhe,Farbe
Beispiel: SLINE 2,4,10,14
Es wird eine senkrechte Linie, Startpunkt (2,4) und Höhe 10 mit der Farbe 14 (hellblau) gezeichnet.

3) RECLOW 49336-495140
Funktion: Zeichnet ein Rechteck in einem LOW-RES-Bildschirm.
Syntax: RECLOW AZ,AS,Breite,Höhe,Farbe



Beispiel: RECLOW 2,3,10,5,1
Zeichnen eines Rechtecks mit dem Startpunkt (2,3 = obere linke Ecke), der Breite 10, der Höhe 5 und der Farbe 1 (weiß).

```
 programm : 4erw .ass          c000 c063
----------------------------------------

c000 : 20 8a ad 20 f7 b7 a5 14  d1
c008 : 85 f7 a5 15 85 f8 20 fd  31
c010 : ae 20 8a ad 20 f7 b7 a5  12
c018 : 14 85 f9 a5 15 85 fa 20  cc
c020 : fd ae 20 8a ad 20 f7 b7  f9
c028 : a5 14 85 9b a5 15 85 9c  fe
c030 : a0 00 a5 01 85 02 78 a9  f7
c038 : 35 85 01 b1 f7 91 9b e6  ef
c040 : f7 d0 02 e6 f8 e6 9b d0  d3
c048 : 02 e6 9c a5 f8 c5 fa d0  e4
c050 : ea a5 f7 c5 f9 d0 e4 a5  c8
c058 : 02 85 01 58 20 79 00 a8  87
c060 : d0 9e 60 3e 03 20 fd ae  e6
```

Listing 8. Modulnummer: 4
Anzahl der Befehle: 1

1) TRANS 49152-49250 0
Funktion: Verschieben eines Speicherbereiches. Dieser Befehl bezieht sich nur auf das RAM, das heißt es kann auch das RAM unter dem Betriebssystem oder Basic-Interpreter benutzt werden.
Syntax: TRANS Staradr.,Endadr.,Startadresse neu
Beispiel: TRANS 1024,2023,40960
Der Bildschirmspeicher wird unter den Basic-Interpreter geschoben.

```
 programm : 5erw .ass          c000 c094
----------------------------------------

c000 : 20 76 c0 20 8a ad 20 f7  16
c008 : b7 20 fd ae 20 9e b7 86  07
c010 : 02 a2 00 20 73 00 f0 06  6e
c018 : 9d 3c 03 e8 d0 f5 a5 01  06
c020 : 85 9e 78 a9 35 85 01 a0  0c
c028 : 00 b1 14 9d 3c 03 e8 c8  cb
c030 : c4 02 d0 f5 a5 9e 85 01  4f
c038 : 58 20 83 c0 20 a5 a9 a5  ba
c040 : 9b 85 7a a5 9c 85 7b 60  96
c048 : 20 76 c0 20 8a ad 20 f7  5e
c050 : b7 20 fd ae 20 28 af 20  af
c058 : 82 b7 85 02 a5 01 85 9b  07
c060 : 78 a9 35 85 01 a0 00 b1  23
c068 : 33 91 14 c8 c4 02 d0 f7  11
c070 : a5 9b 85 01 58 60 a2 00  77
c078 : a9 00 9d 3c 03 e8 e0 c4  95
c080 : d0 f8 60 a5 7a 85 9b a5  27
c088 : 7b 85 9c a9 3c 85 7a a9  4f
c090 : 03 85 7b 60 5b 8a 91 d1  35
```

Listing 9. Modulnummer: 5
Anzahl der Befehle: 2

1) RAMVAR 49152-49223 0
Funktion: Bringt ASCII-Zeichen aus dem RAM in eine Stringvariable.
Syntax: RAMVAR Anfangsadresse,Länge,var$=''
Beispiel: RAMVAR 40960,10,a$=''
Der Inhalt der Speicherzellen 40960-40969 wird in die Variable a$ gebracht. Dieser Befehl bezieht sich nur auf den RAM-Bereich.

2) VARRAM 49224-49269 30
Funktion: Umgekehrte Funktion wie (1). Der Inhalt der Variablen wird in Form der ASCII-Codes ins RAM befördert.
Syntax: VARRAM adresse,variable$
Beispiel: VARRAM 40960,a$
Der Inhalt der Variablen a$ wird ab 40960 ins RAM geschrieben.

```
 programm : 6erw .ass          c000 c0fb
----------------------------------------

c000 : 20 28 af a5 47 85 f7 a5  a1
c008 : 48 85 f8 a0 00 b1 f7 99  06
c010 : 3c 03 c8 c0 05 d0 f6 20  0b
c018 : fd ae 20 28 af a0 00 b1  dd
c020 : 47 91 f7 c8 c0 05 d0 f7  ae
c028 : a0 00 b9 3c 03 91 47 c8  2a
c030 : c0 05 d0 f6 60 20 28 af  8d
c038 : a5 46 48 a5 45 48 20 c4  68
c040 : c0 a9 02 f0 26 a5 f7 85  ee
c048 : f9 a5 f8 85 fa 20 fd ae  09
c050 : 20 28 af 20 c4 c0 a9 02  71
c058 : f0 11 a0 00 68 91 f7 a5  37
c060 : 45 91 f9 c8 68 91 f7 a5  43
c068 : 46 91 f9 60 20 28 af a5  4f
c070 : 46 48 a5 45 48 20 c4 c0  06
c078 : a5 f7 85 f9 48 a5 f8 85  5a
c080 : fa 48 a5 02 f0 30 20 fd  55
c088 : ae 20 28 af 20 c4 c0 a0  b3
c090 : 02 b1 f7 85 9b c8 b1 f7  d0
c098 : 85 9c a0 00 a0 00 b1 f7  54
c0a0 : 91 f9 e6 f7 d0 02 e6 f8  91
c0a8 : e6 f9 d0 02 e6 fa c6 9b  98
c0b0 : d0 ea c6 9c 10 e6 68 85  1f
c0b8 : f8 68 85 f7 68 91 f7 c8  c9
c0c0 : 68 91 f7 60 a6 2f a5 30  d6
c0c8 : 86 f7 85 f8 c5 32 d0 04  03
c0d0 : e4 31 f0 1d a0 00 b1 f7  ed
c0d8 : c8 c5 45 d0 06 a5 46 d1  39
c0e0 : f7 f0 13 c8 b1 f7 18 65  33
c0e8 : f7 aa c8 b1 f7 65 f8 90  4c
c0f0 : d7 a9 00 85 02 60 a9 01  18
c0f8 : 85 02 60 03 20 6d c0 18  97
```

Listing 10. Modulnummer: 6
Anzahl der Befehle: 3

1) SWAPVA 49152-49204 0
Funktion: Vertauschen zweier Variablenwerte.
Syntax: SWAPVA var1,var2
Beispiel: SWAPVA as,k
Die Werte der Variablen as und k werden vertauscht.

2) SWAPAR 49205-49259 0
Funktion: Wie (1), nur für Arrays.
Syntax: SWAPAR arr1,arr2
Beispiel: SWAPAR d$,g$
Der gesamte Inhalt des Array d$(n) wird mit dem des Array g$(n) vertauscht. Es ist zu beachten, daß beide Arrays vorher dimensioniert sein müssen.

3) DUPEAR 49260-49347 55
Funktion: Erzeugt zwei identische Arrays mit angegebenem Namen. Array 1 ist das »Mutterarray«.
Syntax: DUPEAR array1,array2
Beispiel: DUPEAR as,m
Das Array as wird komplett in das Array m übertragen. Es ist darauf zu achten, daß beide Arrays dimensioniert sein und gleiche Größe haben müssen. Anwendung findet dieser Befehl zum Beispiel beim Aufruf von Unterprogrammen, die nachträglich an ein Hauptprogramm angepaßt werden müssen.

```
 programm : 7erw .ass          c000 c00b
----------------------------------------

c000 : a9 e4 8d 08 03 a9 a7 8d  b7
c008 : 09 03 60 a0 00 b1 f7 99  60
```

Listing 11. Modulnummer: 7
Anzahl der Befehle: 1

1) OFF 49152-49162 0
Funktion: Schaltet Erweiterung ab.
Syntax: OFF keine Parameter

```
 programm : 8erw .ass          c000 c153
----------------------------------------

c000 : 20 9e b7 a9 00 48 4c 10  26
c008 : c0 48 20 fd ae 20 9e b7  8a
c010 : 68 a8 8a 99 a7 00 c8 98  71
c018 : c0 05 d0 ed a5 a8 85 d3  aa
c020 : a5 a9 85 d6 20 10 e5 20  30
c028 : 24 ea a5 a7 d0 48 18 a5  1b
c030 : d1 69 28 85 f7 85 f9 a5  4f
c038 : d2 69 00 85 f8 18 a5 f8  48
c040 : 69 d4 85 fa a4 a8 b1 f7  1a
c048 : 91 d1 b1 f9 91 f3 c8 c4  d3
c050 : aa 30 f3 18 a5 d1 69 28  f1
c058 : 85 d1 a5 d2 69 00 85 d2  dc
c060 : 20 24 ea e6 a9 a5 a9 c5  24
c068 : ab d0 c3 85 d6 20 10 e5  97
```

```
c070 : 20 fd ae 4c a4 aa c9 01  8d
c078 : d0 50 e6 aa a5 a8 85 d3  dd
c080 : a5 ab 85 d6 20 10 e5 20  91
c088 : 24 ea 38 a5 d1 e9 28 85  fc
c090 : f7 85 f9 a5 d2 e9 00 85  04
c098 : f8 18 a5 f8 69 d4 85 fa  6e
c0a0 : a4 a8 b1 f7 91 d1 b1 f9  66
c0a8 : 91 f3 c8 c4 aa 30 f3 38  6a
c0b0 : a5 d1 e9 28 85 d1 a5 d2  e0
c0b8 : e9 00 85 d2 20 24 ea c6  b9
c0c0 : ab a5 ab c5 a9 d0 c3 4c  aa
c0c8 : 6b c0 c9 02 d0 3c e6 ab  28
c0d0 : a4 a8 c8 a5 d1 85 f7 18  08
c0d8 : a5 d2 69 d4 85 f8 b1 d1  66
c0e0 : 48 b1 f7 88 91 f7 68 91  ad
c0e8 : d1 c8 c8 c4 aa d0 ef 88  ea
c0f0 : a9 20 91 d1 18 a5 d1 69  11
c0f8 : 28 85 d1 a5 d2 69 00 85  8f
c100 : d2 e6 d6 a5 d6 c5 ab d0  9b
c108 : c7 60 c6 aa e6 ab a5 aa  be
c110 : 85 d3 a5 a9 85 d6 20 10  cd
c118 : e5 a5 d1 85 f7 a5 d2 18  1d
c120 : 69 d4 85 f8 a4 aa 88 b1  99
c128 : d1 48 b1 f7 c8 91 f7 68  52
c130 : 91 d1 88 88 c4 a8 10 ef  8f
c138 : c8 a9 20 91 d1 18 a5 d1  27
c140 : 69 28 85 d1 a5 d2 69 00  ef
c148 : 85 d2 e6 d6 a5 d6 c5 ab  4a
c150 : d0 c7 60 42 03 aa ca 86  22
```

Listing 12. Modulnummer: 8
Anzahl der Befehle: 1

1) SCROLL 49152-49490 0
Funktion: Scrollt rechteckigen Bildschirmausschnitt in beliebige Richtung.
Syntax: SCROLL mode,as,az,es,ez,(var$)
Beispiel: SCROLL 1,2,4,5,10,a$
Hier wird das angegebene Fenster nach unten gescrollt. In die frei gewordene erste Zeile wird der String a$ geprinted.
mo=0 nach oben
mo=1 nach unten
mo=2 nach links
Bei mo=2 und mo=3 entfällt der String am Schluß des Aufrufs.

```
programm : 9erw .ass        c000 c0de
--------------------------------------

c000 : a5 2d 85 f7 a5 2e 85 f8  70
c008 : a0 00 84 c6 b1 f7 10 47  4c
c010 : 29 7f 20 d2 ff c8 b1 f7  58
c018 : 29 7f 20 d2 ff a9 25 20  85
c020 : d2 ff a9 3d 20 d2 ff c8  2e
c028 : b1 f7 85 f9 c8 b1 f7 85  7a
c030 : fa a5 f9 10 18 38 a9 ff  67
c038 : e5 fa 85 fa a9 ff e5 f9  81
c040 : 85 f9 a9 2d 20 d2 ff e6  38
c048 : fa d0 02 e6 f9 a5 f9 a6  0a
c050 : fa 20 cd bd 4c b8 c0 20  53
c058 : d2 ff c8 b1 f7 10 3a 29  cd
c060 : 7f 20 d2 ff a9 24 20 d2  86
c068 : ff a9 3d 20 d2 ff a9 22  a7
c070 : 20 d2 ff a0 02 b1 f7 f0  7d
c078 : 18 85 a7 c8 b1 f7 85 22  8b
c080 : c8 b1 f7 85 23 a0 00 b1  6a
c088 : 22 20 d2 ff c8 c6 a7 d0  72
c090 : f6 a9 22 20 d2 ff 18 90  96
c098 : 1f 20 d2 ff a9 3d 20 d2  26
c0a0 : ff 18 a5 f7 69 02 48 a5  27
c0a8 : f8 69 00 a8 68 20 a2 bb  f3
c0b0 : 20 bc aa a9 91 20 d2 ff  73
c0b8 : 18 a5 f7 69 07 85 f7 a5  96
c0c0 : f8 69 00 85 f8 a9 0d 20  6f
c0c8 : d2 ff a5 f8 c5 30 d0 07  51
c0d0 : a5 f7 c5 2f 30 01 60 a5  a0
c0d8 : c6 f0 fc 4c 08 c0 b1 d1  d0
```

Listing 13. Modulnummer: 9
Anzahl der Befehle: 1

1) DUMP 49152-49373 0
Funktion: Zeigt aktuellen Variablenspeicher auf dem Bildschirm an. Nach jeder Ausgabe kann mit SPACE weitergemacht werden.
Syntax: DUMP keine Parameter

Schwarz auf weiß

Mit den folgenden Änderungen zum Programm »Database« (Sonderheft 7/85) ist es nun auch möglich, Datensätze auf dem Drucker auszugeben. Außerdem können Sie leicht eigene Ideen verwirklichen, zum Beispiel Etikettendruck oder ähnliches.

Um die erforderlichen Änderungen an Database vorzunehmen, gehen Sie bitte wie folgt vor:
1. Bitte laden Sie das Programm »DBII« (Datenpflege).
2. Anschließend geben Sie die in Listing 1 enthaltenen Programmzeilen ein. Bei den Zeilen 330 und 3500 müssen die einzelnen Befehle abgekürzt eingegeben werden (siehe auch Anhang D in der Bedienungsanleitung Ihres C 64).
3. Zum Schluß speichern Sie das geänderte Programm — nachdem Sie die alte Version gelöscht haben — mit »SAVE "DBII", 8« wieder auf Diskette.

Die neuen Kommandos

Neu hinzugekommen sind drei Kommandos im Menüpunkt »Daten ausgeben« (im Teil »Datenpflege«).

Das erste Kommando erreichen Sie durch Drücken der Taste »C«. Mit ihm ist es möglich, den gerade im Speicher befindlichen Datensatz noch einmal einzulesen und auf dem Bildschirm auszugeben. Dabei kann auch zwischenzeitlich in einen anderen Programmteil verzweigt werden.

Ein Beispiel: Sie haben über die Indexsuche einen bestimmten Datensatz selektiert. Nun möchten Sie — aus welchen Gründen auch immer — in den Programmteil »directory ausgeben« verzweigen. Wenn Sie dann anschließend wieder in die Datenausgabe zurückkehren und mit dem zuvor selektierten Datensatz weiterarbeiten wollen, blieb bisher nur die Möglichkeit, diesen nochmals suchen zu lassen. Jetzt drücken Sie einfach die »C«-Taste und der betreffende Datensatz wird wieder in Ihren C 64 geladen und auf dem Bildschirm ausgegeben!

Besonders interessant wird der »C«-Befehl auch im Zusammenhang mit dem zweiten Kommando, das die Ausgabe von Datensätzen auf den Drucker ermöglicht: Durch Drücken der Taste »P« wird die Druckerausgabe ein-, durch Drücken von »O« wieder ausgeschaltet. Außer im Programmteil »Daten ausgeben« kann die Druckerausgabe auch im Hauptmenü (mit denselben Tasten) ein- oder ausgeschaltet werden. Im Hauptmenü wird die eingeschaltete Druckerausgabe durch den Schriftzug »Ausgabe auf Drucker!« (im unteren Bildschirmbereich); im Teil »Daten ausgeben« durch ein »P« in der Kopfzeile (anstelle des »/« nach dem Dateinamen) angezeigt.

Wenn Sie jetzt — bei eingeschalteter Druckerausgabe — einen Datensatz in den Computer laden, so erfolgt die Ausgabe gleichzeitig auf Drucker und Bildschirm! Bei der Druckerausgabe wird dabei jedes Datenfeld in einer neuen Zeile zu Papier gebracht. Am Ende eines Datensatzes werden jeweils zwei Leerzeilen (zur Abtrennung mehrerer Datensätze voneinander) gedruckt.

Um auf den »C«-Befehl zurückzukommen: Wenn Sie sich nicht sicher sind, ob Sie einen bestimmten Datensatz ausdrucken wollen oder nicht, so laden Sie diesen zunächst bei ausgeschalteter Druckerausgabe. Haben Sie sich dann für einen Ausdruck entschieden, schalten Sie jetzt mit »P« die Druckerausgabe ein und drücken anschließend »C« und schon wird der Datensatz ausgedruckt! Danach können Sie mit »O« die Druckerausgabe wieder ausschalten, um nach weiteren Datensätzen zu suchen.

Die Ausgaberoutinen von Database

Wenn Sie selbst Änderungen an Database, speziell an den Ausgaberoutinen, vornehmen wollen, dann werden Ihnen die folgenden Erläuterungen dabei sehr hilfreich sein.

Alle Datensatzausgaben von Database werden zentral von zwei kleinen Unterprogrammen abgewickelt (siehe Listing 2). Dadurch ist es sehr einfach, hier weitgreifende Änderungen vorzunehmen.

Die erste Routine (Zeilen 6355 bis 6397) liest einen Datensatz Datenfeld für Datenfeld in die Variable EG$ ein. Bei einer Datenfeldlänge von mehr als 255 Zeichen wird das Datenfeld in mehreren (bis zu drei) Teilen eingelesen. Ist das Feld zum Beispiel 600 Byte lang, so wird es auf folgende Weise geladen: Zuerst zweimal 255 (= 510) Byte, anschließend die restlichen 90 Byte.

Jedesmal, wenn ein neues Daten(teil)feld in EG$ eingelesen wurde, wird in das zweite Unterprogramm (Zeilen 6415 bis 6478) verzweigt, wo dann die Ausgabe auf dem Bildschirm (oder Drucker) erfolgt.

Diese zweite Routine bietet den besten Ansatzpunkt für Erweiterungen. Wie leicht hier Veränderungen möglich sind, sehen Sie anhand der zusätzlich eingebauten Druckerausgabe.

In Zeile 6415 wird zunächst kontrolliert, ob die Variable XX den Wert 1 enthält. Ist das der Fall, so wird in die Druckerroutine ab Zeile 6475 verzweigt. Ansonsten (und auch nach erfolgter Druckerausgabe) wird mit der alten Routine (zur Bildschirmausgabe) fortgefahren.

Genausogut wie in die Zeile 6475 verzweigt wird, kann natürlich zu jeder beliebigen anderen Stelle gesprungen werden! So wäre es zum Beispiel ohne weiteres denkbar (auch zusätzlich zur gleichzeitig stattfindenden Bildschirm-Druckerausgabe), Datensätze in eine sequentielle Datei schreiben zu lassen, um sie dann an anderer Stelle in einer Textverarbeitung zur Serienbrieferstellung weiterzuverarbeiten.

Das Prinzip zum Schreiben in eine sequentielle Datei ist im Grunde genommen dasselbe wie zur Ausgabe auf den Drucker. Schauen wir uns also die Druckerroutine etwas genauer an.

In Zeile 6475 wird eine Datei auf den Drucker (Geräteadresse 4) eröffnet.

In Zeile 6476 wird durch den Vergleich von S6 und G1+1 ermittelt, ob es sich bei dem aktuellen Inhalt von EG$ um ein komplettes Datenfeld oder um ein Datenteilfeld handelt. Ist S6 nämlich ungleich G1+1, dann befindet sich in EG$ ein Datenteilfeld (eines Datensatzes größer als 255 Zeichen). In diesem Fall wird der Wagenrücklauf am Ende des Ausdrucks durch das Semikolon hinter EG$ verhindert. Dadurch wird erreicht, daß das nächste Daten(teil)feld direkt dahinter ausgedruckt wird. Im gegenteiligen Fall, also wenn S6 = G1+1, enthält EG$ ein komplettes Datenfeld beziehungsweise das letzte Teilfeld eines größeren Datenfeldes. In diesem Fall wird am Ende des Ausdrucks ein Wagenrücklauf erzeugt.

In Zeile 6478 wird dann die Datei wieder geschlossen und zur Aufrufstelle zurückgekehrt.

Um eine neue Funktion benutzen zu können, muß diese natürlich eingeschaltet (und möglichst auch ausgeschaltet) werden können. Wie das aussehen könnte, sehen Sie in den Zeilen 215 und 216 beziehungsweise 3029 und 3030 (Listing 1). Die Variable EG enthält jeweils den ASC-Code der gedrückten Taste. Entsprechend dieser Taste wird dann die Variable XX, die ja bei der Datensatzausgabe kontrolliert wird, auf 0 oder 1 gesetzt (und eventuell eine Meldung auf dem Bildschirm ausgegeben).

Noch ein Tip: Nähere Informationen zu allen verwendeten Variablen finden Sie in den Tabellen 5 und 6 im Artikel zu Database (Sonderheft 7/85, Seite 94/95). Bei allen Änderungen sollten Sie jedoch folgendes beachten:

Kleinere Ergänzungen, wie zum Beispiel einen Etikettendruck, für den ja an sich nur druckerspezifische Steueranweisungen eingefügt werden müssen, können ohne weiteres direkt in die Ausgaberoutine integriert werden. Bei umfangreicheren Erweiterungen sollten Sie allerdings auf einen neuen Programmteil ausweichen, da der für die Variablen zur Verfügung stehende Speicherplatz ohnehin sehr knapp ist und jeder hinzukommende Basictext die Maximalanzahl der Datenfelder weiter herabsetzt.

Zur Erinnerung: Ein zusätzliches Programm kann ohne weiteres in das Gesamtsystem integriert werden, vorausgesetzt es erfüllt zwei Bedingungen: Es ist, zumindest teilweise, ein Basicprogramm, und es ist unter dem Namen »SOFT« auf Diskette gespeichert!

Im Lademenü kann dieses Programm dann über den Menüpunkt »Sonderfunktionen« (»F5«-Taste) aufgerufen werden.

Zum Schluß noch ein Wort zur Druckerausgabe:

Wie Sie sicher schon bemerkt haben, ist bei der Dateneingabe jedes Datenfeld mit einem Sonderzeichen (einem kleinen Punkt) vorbelegt. Wenn Sie bisher ein Datenfeld nicht in seiner ganzen Länge benötigten, konnten Sie den Restbereich getrost mit diesem Zeichen aufgefüllt lassen. Bei der Druckerausgabe wird aber nun natürlich nicht dieses Sonderzeichen, sondern das Originalzeichen (ein Grafikzeichen (CHR$(166)) gedruckt (was nicht gerade zur optischen Aufbesserung des Ausdrucks beiträgt). Um das zu vermeiden, sollten Sie bei der Dateneingabe oder nachträglich über die Änderungsfunktion (dazu drücken Sie bei der Datenausgabe die Taste »R«) alle Restbereiche mit einem Leerzeichen auffüllen.

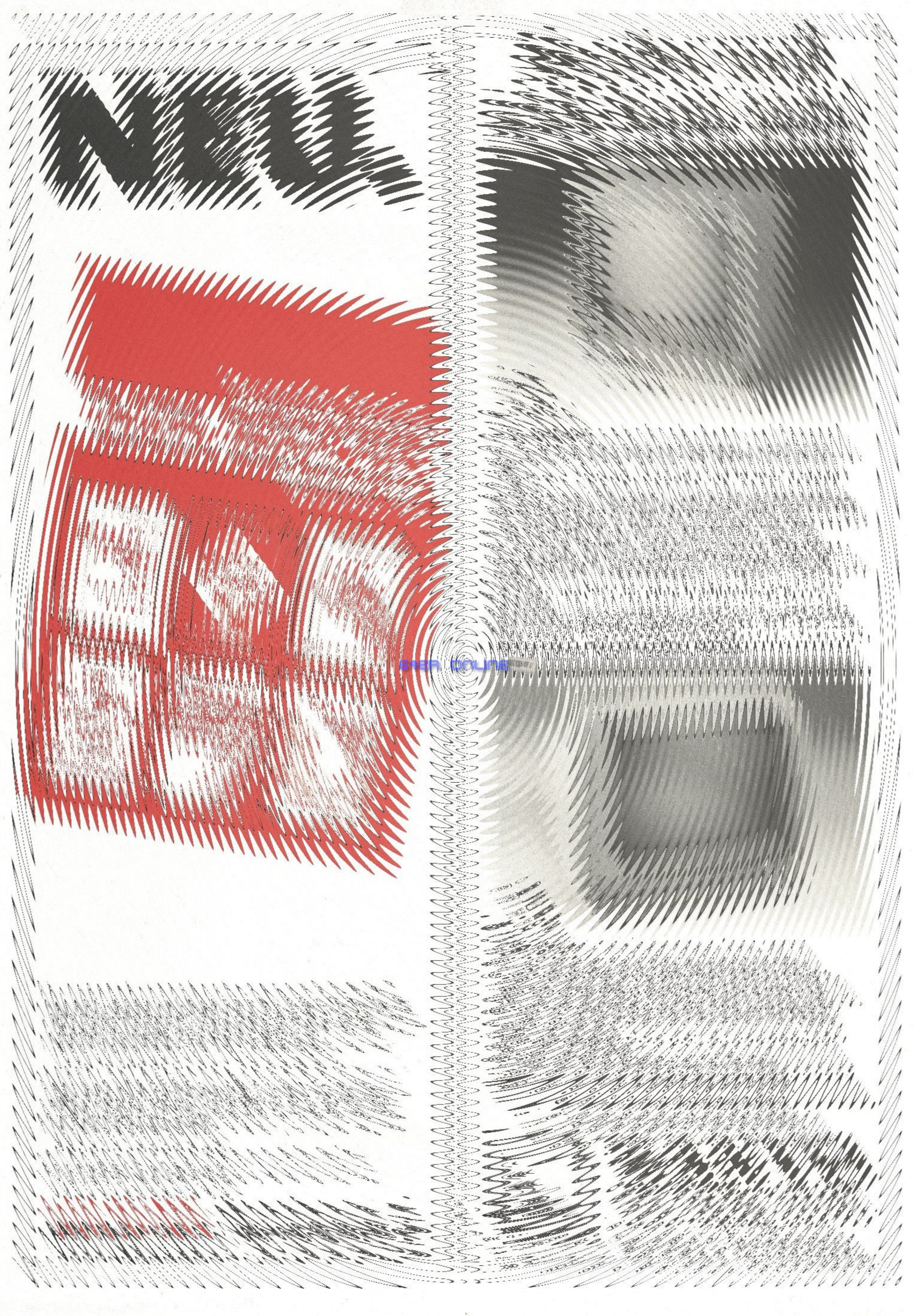
(Martin Hecht/tr)

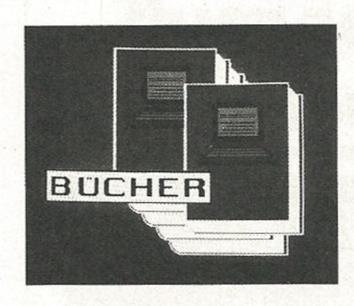
```
100 REM--- ERGAENZUNGEN ZU DATABASE ------
    ------------------------------------ <101>
105 REM--- (W) 1985 BY MARTIN HECHT,STGT -
    ------------------------------------ <127>
110 :                                    <086>
182 IF XX=1 THEN ZE=21:SP=10:GOSUB 11050:P
    RINT"{RVSON}AUSGABE AUF DRUCKER!{RVOFF
    }"                                   <124>
215 IF EG=80 THEN XX=1:ZE=21:SP=10:GOSUB 1
    1050:PRINT"{RVSON}AUSGABE AUF DRUCKER!
    {RVOFF}"                             <050>
216 IF EG=79 THEN XX=0:ZE=21:SP=10:GOSUB 1
    1050:PRINT"{20SPACE}"                <071>
330 FOR S1=20 TO 5 STEP-1:ZE=S1:SP=5:GOSUB
     11050:PRINT KS$(110):NEXT S1:GOSUB 40
    5:POKE 2004,K:RETURN                 <238>
3016 IF XX=1 THEN ZE=0:SP=24:GOSUB 11050:P
     RINT"{RVSON}P{RVOFF}";              <116>
3026 GOSUB 11455:IF EG=8 THEN GOSUB 6131:G
     OTO 3030                            <224>
3029 IF EG=80 THEN XX=1:ZE=0:SP=24:GOSUB 1
     1050:PRINT"{RVSON}P{RVOFF}";        <253>
3030 IF EG=79 THEN XX=0:ZE=0:SP=24:GOSUB 1
     1050:PRINT"{RVSON}/{RVOFF}";        <040>
3031 IF EG=67 THEN GOSUB 3500            <119>
3032 FOR S1=41 TO 56:IF FC%(S1)<>EG THEN N
     EXT S1:GOTO 3026                    <077>
3500 IF T3=-1 THEN GOSUB 9782:ZE=23:AG$=KS
     $(52):GOSUB 11585:POKE 2,1:SYS 51833:
     GOSUB 9782:RETURN                   <051>
3501 GOSUB 3171:RETURN                   <134>
6390 CLOSE 1:GOSUB 9782:DC=1             <234>
6391 IF XX=1 THEN OPEN 4,4:PRINT#4:PRINT#4
     :CLOSE 4                            <121>
6392 RETURN                              <098>
6415 IF XX=1 THEN GOSUB 6475             <000>
6416 IF S6<>1 THEN 6435                  <182>
6475 OPEN 4,4                            <228>
6476 IF S6<>G1+1 THEN PRINT#4,EG$;:GOTO 64
     78                                  <251>
6477 PRINT#4,EG$                         <207>
6478 CLOSE 4:RETURN                      <023>
```

Listing 1. Diese Zeilen müssen Sie zum Programmteil »DBII« zusätzlich eintippen.

```
6350 REM--- DATENSATZ EINLESEN ----------------
     ----------------------------------
6352 :
6355 GOSUB 9782:PG=ES:GOSUB 6010:ZE=23:SP=7:GOS
     UB 11050:PRINT KS$(49)C$C$:S2=0
6356 OPEN 1,8,2,DD$(SI)+".R,L,"+CHR$(254)
6357 PRINT#15,"P"+CHR$(2)+CHR$(PO(1))+CHR$(PO(2
     ))+CHR$(PO(3))
6365 FOR S1=1 TO AF:G1=INT(FD(S1,2)/255):G2=FD(
     S1,2)-G1*255
6367 FOR S6=1 TO G1+1:L1=255-PO(3):IF S6<>G1+1
     THEN H4=255:GOTO 6375
6370 H4=G2
6375 IF L1>H4 THEN QX=H4:GOSUB 6905:GOSUB 6415:
     PO(3)=PO(3)+H4:NEXT S6:NEXT S1:GOTO 6390
6380 IF L1=H4 THEN QX=H4:GOSUB 6905:GOSUB 6415:
     GOSUB 6396:NEXT S6:NEXT S1:GOTO 6390
6382 POKE 2,L1:POKE 69,90:POKE 70,215:SYS 52453
     :GOSUB 6396:QX=H4-L1:GOSUB 6905:EG$=ZW$+EG
     $:GOSUB 6415
6385 IF H4-L1=254 THEN GOSUB 6396:NEXT S6:NEXT
     S1:GOTO 6390
6387 PO(3)=PO(3)+H4-L1:NEXT S6:NEXT S1
6390 CLOSE 1:GOSUB 9782:DC=1
6391 IF XX=1 THEN OPEN 4,4:PRINT#4:PRINT#4:CLOS
     E 4
6392 RETURN
6396 PO(3)=1:PO(1)=PO(1)+1:IF PO(1)=256 THEN PO
     (1)=0:PO(2)=PO(2)+1
6397 PRINT#15,"P"+CHR$(2)+CHR$(PO(1))+CHR$(PO(2
     ))+CHR$(PO(3)):RETURN
6400 :
6401 :
6402 :
6405 REM--- DATEN(TEIL)FELD AUSGEBEN ----------
     ----------------------------------
6407 :
6415 IF XX=1 THEN GOSUB 6475
6416 IF S6<>1 THEN 6435
6417 S2=S2+1:IF S2<=AF%(PG)THEN 6435
6420 GOSUB 11243
6425 PG=PG+1:IF AF%(PG)=0 THEN 6425
6430 S2=1:GOSUB 6010
6435 IF S6=1 THEN G1=INT(FD(S1,2)/255):H1=FD(S1
     ,1)-50176:ZE=INT(H1/40):SP=H1-40*ZE
6445 GOSUB 11050:PRINT EG$
6450 IF S6<>G1+1 THEN H1=H1+255:ZE=INT(H1/40):S
     P=H1-40*ZE
6470 IF S1=AF AND S6=G1+1 THEN GOSUB 11243:PG=E
     S:GOSUB 6010
6472 RETURN
6475 OPEN 4,4
6476 IF S6<>G1+1 THEN PRINT#4,EG$;:GOTO 6478
6477 PRINT#4,EG$
6478 CLOSE 4:RETURN
```

Listing 2. Dient nur der Erläuterung des Textes und braucht nicht abgetippt zu werden.





C 64 FISCHERTECHNIK

Messen — Steuern — Regeln, ein hochaktuelles Thema durch die computergesteuerten Produktionsanlagen in der industriellen Fertigung. Mit dem Fischertechnik Computing-Baukasten eröffnen sich auch für den Homecomputer neue Anwendungsmöglichkeiten auf diesem Bereich. Das Buch »C 64 Fischertechnik« führt den Leser anhand von vielen Modellen gut in diese Thematik ein.

Zunächst wird schrittweise der Umgang mit den Ein- und Ausgängen des Interface erläutert. Der Autor beschreibt auch, wie Sie diese mit Ihrem Computer steuern können.

Im folgenden Kapitel »BASIC-Programmierung des Interface« erklärt der Autor zunächst, wie Sie einzelne Bauelemente (Magneten, Motoren, Lampen, etc.) ein- und ausschalten oder bestimmte Drehrichtungen vorgeben. Wie Sie in Ihrem Basic-Programm Schalterstellungen oder Positionen eines Schwenkarms abfragen können, erfahren Sie anschließend.

An immer komplexeren Beispielen wendet der Autor nun die bisherigen Grundkenntnisse an. Dabei werden verschiedene Modelle besprochen — angefangen von einer Sortieranlage, weitergehend über ein automatisches Längenmeßgerät und verschiedenen Greifarmen bis hin zu einem Minischach-Roboter. Der Autor Said Baloui stellt die notwendigen Programme in Teilschritten vor und erklärt diese ausführlich. Neulinge auf dem Gebiet Messen — Steuern — Regeln haben daher auch kaum Probleme, sich anhand des Buches so weit in die Materie einzuarbeiten, daß sie anschließend eigene Steuerprogramme schreiben können.

Im letzten Abschnitt des Buches kommen die Maschinensprache-Fans auf Ihre Kosten. Anhand eines kleinen Videospiels, dessen Steuerung mit zwei Potentiometern von Fischertechnik erfolgt, lernt der Leser wichtige Maschinensprache-Routinen kennen. Danach kommt Said Baloui zu der eigentlichen Interfacesteuerung in Assembler. Ähnlich dem Basic-Kapitel werden dabei einzelne Programmroutinen vorgestellt und ausführlich besprochen. Nun lassen sich auch besonders zeitkritische Aufgaben lösen.

Insgesamt stellt das Buch eine sinnvolle Erweiterung zu der Anleitung dar, die dem Fischertechnik Computing Baukasten beiliegt. Besonders, wer dabei völliges Neuland betritt, wird dieses Buch als hilfreiche Stütze schätzen lernen. (kn)

Info: Said Baloui, »C 64 Fischertechnik, Messen — Steuern — Regeln«, Markt & Technik 1986, ISBN 3-89090-194-8, Preis 29,90 Mark

NUMERISCHE MATHEMATIK

Dieses Buch ist der mittlerweile vierte Band aus der Reihe »Anwendung von Mikrocomputern«, in der der Vieweg-Verlag Programmpakete und Problemlösungen zu Themen aus den Bereichen Betriebswirtschaft, Naturwissenschaft und Technik anbietet. Da die numerische Mathematik eher als Stiefkind weiterführender Schulen und ingenieurwissenschaftlicher Studiengänge behandelt wird, dürfte dieses Buch vornehmlich Mathematiker und Informatiker und solche, die es werden wollen, interessieren.

Es stehen insgesamt 40 Basic-Programme zur Verfügung, die auf jeden Mikrocomputer übertragbar sind, da auf spezielle Basic-Befehle verzichtet wurde. Bei der Auswahl der numerischen Verfahren wurden hinsichtlich der Effektivität und der Genauigkeit die Eigenarten von Mikrocomputern berücksichtigt. Es wurden vielfach neue, wenn auch weniger bekannte Verfahren benutzt. Als Beispiel sei hier die Integration durch rationale Extrapolation nach Stoer genannt, die, kombiniert mit der Schrittweitenhalbierung nach Bulirsch, eine der effektivsten Integrationsmethoden darstellt. Zu jedem Problemkreis werden verschiedene Methoden angeboten und ausführlich erläutert. Außer durch die jeweiligen Formeln werden die meisten Verfahren in Flußdiagrammen wiedergegeben. Die Programme selbst bestehen durch einen übersichtlich strukturierten Programmierstil, womit auch deren Verwendung in Subroutinen einfach wäre.

Die Bibliothek umfaßt Themen wie Funktionsdarstellung, nichtlineare und lineare Gleichungen, Fehler- und Ausgleichsrechnung, Interpolation, Differentiation und Integration, Eigenwertprobleme sowie gewöhnliche und partielle Differentialgleichungen.

Fazit: Dieses Buch kann durch sein didaktisches und inhaltliches Konzept sehr überzeugen und ist jedem wärmstens zu empfehlen, der sich in höherer Mathematik zu Hause fühlt.

(Matthias Rosin/ev)

Info: Dietmar Herrmann, Numerische Mathematik, Vieweg-Verlag, ISBN 3-528-04299-0, 140 Seiten, Preis: 29,80 Mark

C 1571 & 1570: DAS GROSSE FLOPPYBUCH

Das im Buchtitel enthaltene Prädikat »groß« verdient dieses Buch von Data Becker sicherlich, denn mit 583 Seiten ist es ein umfangreiches Werk.

Die große Stärke des Buches ist das rund 350 Seiten umfassende DOS-Listing, das zudem eine Cross-Reference enthält. Es ist sehr gewissenhaft dokumentiert und glänzt durch gute optische Aufmachung. Hier hat man sich ganz offensichtlich an der ausgezeichneten Darstellung in Karsten Schramms Buch »Die Floppy 1541« orientiert, was dem Buch sehr zugute kommt.

Gegen das DOS-Listing und die erfreulichen Tabellen fällt der Rest des Buches stark ab; der Autor schreibt zwar, er bemühe sich um Verständlichkeit, doch schon die unglaublich zahlreichen Verstöße gegen die Rechtschreibung (insbesondere die Zeichensetzung) erschweren die Lektüre des Buches unnötig. Außerdem finden sich kaum Beispielprogramme, und wenn eines abgebildet ist, so kann man es meistens nur brauchen, wenn man mit IBM-System-34-Disketten arbeitet (!). Vor allem das komplexe Thema »relative Dateien« wird viel zu theoretisch beschrieben.

Ein wirklich brauchbares Beispielprogramm ist ein Disketten-Monitor, der in der Tat hervorragend ist. Leider ist das Listing nicht im Buch abgedruckt, sondern nur für weitere 30 Mark auf Diskette erhältlich. Ohne den Disk-Monitor aber ist ein Großteil des Buches wertlos, man muß sich also wohl oder übel auch die Diskette bestellen. So kommt man in eine Preislage, die eindeutig zu hoch ist.

(Florian Müller/ev)

Info: Rainer Ellinger, C1571&1570: Das große Floppybuch, Data Becker, 583 Seiten, ISBN 3-89011-124-6, Preis: 69 Mark, Diskette mit Beispielprogramm: 30 Mark

PROFESSIONELLE SOFTWARE FÜR DEN COMMODORE 64

Hier liegt einmal mehr ein Buch vor, das beweist, daß der Commodore 64 mehr Qualitäten besitzt als ein einfacher Spielcomputer. Geboten wird eine Sammlung von Programmen aus mathematischen und technischen Bereichen. Der Begriff »Professionell« ist für mein Empfinden nicht zu hoch gegriffen. Der übersichtliche Programmierstil und die angemessene Art der Kommentierung ermöglichen ein schnelles Nachvollziehen der Programme. Der Inhalt beginnt mit einem weitgefaßten Spektrum mathematischer Anwendungen.

Aus der angebotenen Themenfülle seien hier nur einige Beispiele genannt, wie Fouriertransformation, Determinanten und Matrizen, Integral- und Differentialrechnung, Vektor-Operationen, Komplexe Zahlen, Wahrscheinlichkeitsverteilungen, Kombinatorik etc. Ein anderes Kapitel widmet sich allgemeinen Problemen der Elektrotechnik. Geboten werden hier Programme zur Berechnung von Dämpfungsgliedern und Schwingkreis-Impedanzen. Ein besonderes Augenmerk möchte ich auf zwei Programme lenken: Es wird ein Textverarbeitungsprogramm vorgestellt, das mit einigen professionellen Besonderheiten aufwarten kann, wie zum Beispiel einen 80-Zeichen-Scrolling oder Speicherung auf Disk oder Kassette.

Ein echter Leckerbissen ist das folgende Simulationsprogramm: Durch kybernetische Verknüpfung analoger Funktionsblöcke kann man ein Simulationsnetzwerk aufbauen und so natürliche Vorgänge simulieren, die sich mathematisch nur in aufwendigen Differentialgleichungen darstellen lassen. Als praktisches Beispiel wird der Vorgang der gedämpften Federschwingung analysiert.

Das Kapitel über Spiele wird dem Anspruch »auch ein Profi muß mal abschalten« (gottlob) nicht gerecht, da es sich um einige kleine raffinierte Strategiespiele handelt, die die grauen Hirnzellen zusätzlich strapazieren. Das abschließende Kapitel über Assemblerprogrammierung ist in diesem Buch eher fehl am Platz, da der mittelmäßige Basic-Assembler sowie der »Maschinenkursus« auf acht Seiten nicht überzeugen können. Optisch enttäuschend waren auch die teilweise unleserlichen Listings.

Der Gesamteindruck jedoch läßt dieses Buch empfehlenswert für jedermann erscheinen, der sich intensiv mit den angesprochenen Themen auseinandersetzen will, insbesondere für Oberstufenschüler und Studenten.

(Matthias Rosin/ev)

Info: Dipl.-Ing. K.L. Boon/Dipl.-Ing. A.R. Pelsmaeker, Professionelle Software für den Commodore 64, Addison-Wesley Verlag, 230 Seiten, ISBN 3-925118-01-2, Preis: 54 Mark

Auf der Suche nach der Grafik

Egal, wo Ihre HiRes-Grafik liegt, »Hardmaker« findet sie. Multi-Color-Grafiken können auf Wunsch in Graustufen umgerechnet und ausgedruckt werden.

Der »Hardmaker« erlaubt es, HiRes-Grafiken aus fast allen Programmen aufs Papier zu bringen. Dazu stehen umfangreiche Routinen zur Verfügung, die dem Benutzer fast alle Arbeiten abnehmen. Nur das Papier müssen Sie noch von Hand einspannen.

Eingabehinweise: Geben Sie den »Hardmaker« (siehe Listing) bitte mit dem MSE ein, und speichern Sie ihn. Der Programmstart erfolgt durch RUN.

Danach sehen Sie auf dem Bildschirm (meistens) ein wüstes Durcheinander von Punkten. Das ist der Bereich von $2000 bis $4000 als Multicolor-Grafik dargestellt. Das könnte zum Beispiel Teil eines Computerspiels sein, das vorher im Computer war. Wenn Sie nun ein Programm auf Bilder untersuchen wollen, müßten Sie im »Diskworkmodus« dieses Programm laden, die Grafik finden, eventuell speichern und ausdrucken. Dazu stehen Ihnen folgende Funktionen zur Verfügung.

Speicherbereiche:

Computergrafiken können nur an bestimmten Stellen im Speicher stehen, um vom VIC ausgelesen werden zu können. Ein solcher Bereich ist der von $2000 bis $3FFF. Diesen Bereich sehen Sie grundsätzlich auf dem Bildschirm; er wird vom Programm als Grafik-RAM benutzt. Wollen Sie den Inhalt eines anderen Bereiches sehen, muß er nach $2000 transportiert werden. Dazu dienen die Tasten 1 bis 6 und ←:

1:	$4000 bis $5FFF
2:	$6000 bis $7FFF
3:	$8000 bis $9FFF
4:	$A000 bis $BFFF (RAM unterm Basic)
5:	$C000 bis $DFFF ($D000 bis $DFFF; RAM unter I/O)
6:	$E000 bis $FFFF (RAM unterm Kernel)
←:	$0000 bis $1FFF, Dieser Bereich ist nur der Vollständigkeit halber per Taste erreichbar. Benutzen können Sie ihn nicht, da dort Zeropage, Stack, Video-RAM und der Hardmaker selbst liegen!

Wenn Sie auf eine dieser Tasten ohne SHIFT, CTRL oder CBM drücken, wird der entsprechende Speicherbereich nach $2000 transportiert und ist damit auf dem Bildschirm sichtbar. Drücken Sie jedoch SHIFT und dann eine dieser Tasten, wird der entsprechende Bereich mit dem ab $2000 ODER-verknüpft. So können zwei Bilder zusammengemischt werden. Das Ergebnis liegt wieder ab $2000.
Folgende Kombinationen bewirken also:

a) (nur Taste ← bis 6)	≙ Bereich nach $2000 transportieren
b) SHIFT	≙ ODER-Verknüpfen
c) CBM	≙ EX-OR-Verknüpfen
d) SHIFT und CBM	≙ UND-Verknüpfen
e) CTRL	≙ Bereich mit dem ab $2000 vertauschen
f) CTRL SHIFT (oder CTRL C=)	≙ $2000 bis $3FFF in entsprechenden Bereich kopieren

Wenn Sie eine Kombination mit CTRL drücken, wird der entsprechende Speicherplatz verändert! Auf diese Weise können Sie zum Beispiel den Inhalt von $2000 bis $3FFF zwischenspeichern, wenn Sie ihn danach weiterbearbeiten wollen (Zum Beispiel bei schwierigen Korrekturen).

Bilder »schneiden«

Manchmal kommt es vor, daß nicht der gesamte Inhalt des Bildschirms zu einer Grafik gehört und man den Rest »wegschneiden« möchte. Zum Beispiel möchten Sie am rechten Rand etwas entfernen. Dazu drücken Sie die Taste R. Der Rahmen wechselt seine Farbe und an der rechten Seite erscheint eine flackernde Linie, die Sie mit Cursor links/rechts hin- und herbewegen können.

Wenn Sie die Linie richtig positioniert haben, drücken Sie auf SPACE; der Bereich rechts der Linie wird gelöscht (oder gefüllt, wenn sie SHIFT SPACE drücken). Möchten Sie nichts löschen, dann drücken Sie Q, und die flackernde Linie ist verschwunden.

r:	rechts	Randfarbe: orange
l:	links	Randfarbe: blau
o:	oben	Randfarbe: hellrot
u:	unten	Randfarbe: grün

Im Schneidemodus stehen folgende Funktionen zur Verfügung:
— Cursor nach oben/unten (nur o, u)
— Cursor nach oben links/rechts (nur l, r)
— SPACE, SHIFT-SPACE
— Q (wie quit)

Bilder verschieben:

Wenn die Grafik nicht genau oben links beginnt, muß sie verschoben werden. Eine Möglichkeit dazu ist das Scrollen:

Mit Commodore-Taste + Cursor-Taste wird die Grafik um 1 Byte nach links oder rechts verschoben.

Die andere Möglichkeit sind die Tasten A und E. Positionieren Sie den Cursor irgendwo mitten auf dem Bildschirm und drücken Sie die Taste A: Die Grafik wird so verschoben, daß die Cursorposition nun den Anfang der Grafik bildet. Analog funktioniert hier die Taste E: Die Cursorposition wird zum Ende der Grafik. Gerade diese Funktionen ermöglichen ein bequemes Positionieren einer Grafik, die irgendwo im Speicher liegt.

Farbe

Mit den Funktionstasten kann die Farbe der Grafik geändert werden, wenn die Ausgangsbelegung (die sich für Schwarzweiß-Ferseher übrigens gut eignet) nicht gefällt:

f1/f2	≙	Farbe 1 HiRes/Multi
f3/f4	≙	Farbe 2 HiRes/Multi
f5/f6	≙	Farbe 3 Multi
f7/f8	≙	Farbe 4 Multi

Die Funktionstaste blättert die Farben vorwärts, geshiftet blättert sie die Farben zurück. Für Multi und HiRes sind getrennte Farb-Speicher vorhanden.

Folgende Tasten bewegen den Cursor:

Cursor links/rechts /oben/unten	gewohnte Cursor-Bewegung
RETURN	Bewegt ihn in die erste Spalte der nächsten Zeile
SHIFT HOME (CLEAR)	löscht die Grafik
HOME	Cursor links oben
£	(Gegenteil von HOME: Cursor in letzte Spalte letzte Zeile)
SPACE	Cursor rechts + Cursorfeld löschen
SHIFT SPACE	Cursor rechts + Cursorfeld füllen mit Farbe 3
CBM SPACE	Cursor rechts + Cursorfeld füllen mit Farbe 1
CTRL SPACE	Cursor rechts + Cursorfeld füllen mit Farbe 2
DEL	genauso wie SPACE, bloß mit Cursor nach links (also kein echtes DEL!)

Sonderfunktionen:

H	schaltet HiRes-Modus ein; genauso wie Sie die Grafik nun sehen, wird sie von einem Matrixdrucker ausgegeben werden.
M	schaltet Multicolor an.

T ist eine sehr praktische Sache, wenn man einen der erwähnten Matixdrucker besitzt. Diese geben die Multicolor-Grafiken nämlich so aus, daß die Farben 1 und 2 als charakteristische Linien erscheinen. T verwandelt diese Farben nun in Graustufen, die der Drucker ausgeben kann (siehe Kasten).
T funktioniert nur, wenn Multicolor eingeschaltet ist, und schaltet dann auf HiRes um!

I invertiert die Grafik

S spiegelt die Grafik an der Vertikalen, und vertauscht die Farben 1 und 2, die ja auch gespiegelt werden. Spiegeln an der Horizontalen ist nicht nötig, da man ja die Hardcopy einfach umdrehen kann!

X kehrt ins Basic zurück.

Diskworkmodus:

Wird mit D aktiviert. Die Grafik wird dann ausgeblendet und Sie haben den normalen Kernel-Editor vor sich. Folgende Befehle wurden hier implementiert.

$	Directory
@	Kommandokanal des Laufwerks abfragen
@i	initialisieren (analoges gilt für SCRATCH, RENAME, FORMAT, etc.)
n	gibt die aktuelle Geräteadresse aus (Voreingestellt 8)
n9	schaltet auf Geräteadresse 9 um
?	ist der normale Basic-PRINT-Befehl, der hier für Berechnung genutzt werden kann
Q	schaltet den Grafikmodus wieder an
X	Rückkehr zum Basic
←	Mit ← kann aus dem Hardmaker heraus eine Grafik gespeichert werden; zum Beispiel »← Bild 1 (return)« speichert den Bereich $2000 bis $3FFF (genauer bis $3F40) auf Diskette unter dem Namen »Bild 1« ab. Das erzeugt Programmfile ist 32 Blöcke lang.
£	Um ein Programm nach Grafiken zu durchsuchen, muß man es irgendwie in den Computer bekommen. Die eine Möglichkeit ist, das Programm zu laden, zu starten, mit Reset auszusteigen und dann den Hardmaster nachzuladen. Dies empfiehlt sich immer dann, wenn das Programm die Grafik erst erzeugt. Außerdem liegt diese dann schon meist an der richtigen Stelle im Speicher, so daß man nicht mehr zu verschieben braucht. Die andere Möglichkeit ist, erst den Hardmaker zu laden und dann mit dem £-Befehl das betreffende Programm. Dies wird dann in den Bereich ab $2000 geladen, so daß ein Autostart entfällt. Diese Methode kann zu Problemen führen, wenn das betreffende File länger als 178 Blöcke ist, weil dann in die Register der I/O-Bausteine hineingeladen wird. Diese Methode ist dann nötig, wenn ein Programm sich nicht durch Reset stoppen läßt.

Hardcopy

Dieser Programmteil ist zugleich der wichtigste wie auch der problematischste. Denn eine Hardcopy-Routine muß an die meisten Drucker speziell angepaßt werden. Die im »Hardmaker« integrierte Hardcopy-Routine spricht Drucker mit sieben Nadeln an, also den MPS 801, den MPS 803, und Epsondrucker, deren Interface eine Umwandlung auf acht Nadeln vornehmen kann. Hier hilft entweder nur ausprobieren, oder eine andere Hardcopy-Routine. Diese kann ab der Adresse $13A0 an den »Hardmaker« angehängt werden.

Aktiviert wird die Hardcopy mit der Taste P. Daraufhin färbt sich der Rahmen schwarz und der C 64 beginnt mit der Hardcopy. Danach wird in den Grafikmodus zurückgesprungen.

(Christian Kurts/og)

Funktionsweise des T-Befehls beim Hardmaker

Im Multicolor-Modus stellen immer je 2 Bit eines Bytes die Information für einen Bildpunkt zur Verfügung, so daß die Auflösung gegenüber dem HiRes-Modus halbiert wird. Dafür hat man vier Farben statt zwei zur Auswahl, die durch die Bitkombinationen 00, 01, 10 und 11 repräsentiert werden. Ein normaler Matrixdrucker hat hingegen keinen Multicolor-Modus und stellt die Bitmuster genau so dar, wie sie im Grafik-ROM stehen. Das führt bei Gebieten, die aus den Farben 1 und 2 beziehungsweise den Bitmustern 01 oder 10 bestehen, zu den bekannten Streifenmustern (Bild 1). Um diese in Graustufen (Bild 2) umzuwandeln, geht der Computer folgendermaßen vor: Die vier Zweierbit-Kombinationen eines jeden der 8000 Byte des Grafik-RAMs werden nach den Bitkombinationen 01 und 10 untersucht.

1. Fall: Zweierbit-Kombination = 00 oder 11: nichts wird verändert.

2. Fall: Zweierbit-Kombination = 01: In diesem Fall werden die zwei Bit invertiert, also durch 10 ersetzt, aber nur dann, wenn die zwei Bit Bestandteil des 1., 3., 5., etc. Bytes des Grafik-RAMs sind. Dadurch wird das Bitmuster in jeder zweiten Grafikzeile um ein Bit nach links versetzt. Wenn die ersten 3 Byte also

```
01 01 00 01                          10 10 00 10
01 01 11 01  lauten, so wird daraus  01 01 11 01
01 01 01 01                          10 10 10 10
```

und der Drucker gibt ein gleichmäßig graues Feld aus.

3. Fall: Bitkombination = 10

Hierbei würde aus den ersten 3 Byte

```
10 10 00 10                    01 10 00 10
10 10 11 10  folgendes werden: 10 01 11 01
10 10 10 10                    01 10 01 10
```

Also ein etwas gröberes Graufeld. Dieser Effekt wird dadurch erreicht, daß eine Zweierbit-Kombination nur dann invertiert wird, wenn sie entweder im 1., 3., 5., ... Byte die 1. oder 3. Zweierkombination oder im 2., 4., 6., ... Byte die 2. oder 4. Zweierkombination ist.



Bild 1. Eine Hardcopy ohne Umrechnung...



Bild 2. ...und eine mit Umrechnung der Multi-Color-Daten.

```
programm : hardmaker          0801 1500
-------------------------------------

0801 : 25 08 c0 07 9e 20 32 30   4f
0809 : 38 37 20 20 09 0e 05 57   ad
0811 : 52 49 54 54 45 4e 20 42   73
0819 : 59 20 43 48 52 2e 4b 55   cb
0821 : 52 54 53 00 00 00 a9 0f   37
0829 : 85 fc 8d de 03 a0 00 8c   ba
0831 : dd 03 a9 cb 85 fb a9 20   93
0839 : 8d e0 03 a9 00 8d df 03   1e
0841 : a9 e8 8d 00 03 a9 10 8d   9b
0849 : 01 03 a9 08 8d e2 03 a9   86
0851 : 80 85 9d ad 11 d0 29 bf   6d
0859 : 09 38 8d 11 d0 ad 16 d0   78
0861 : 09 18 8d 16 d0 ad 18 d0   19
0869 : 09 08 8d 18 d0 20 4d 09   32
0871 : a9 13 20 d2 ff a9 0f 8d   ab
0879 : 20 d0 20 14 0b 85 40 8a   7f
0881 : 18 69 20 85 41 a2 02 a0   79
0889 : 07 b1 40 49 ff 91 40 88   41
0891 : 10 f7 ca d0 f2 a5 c6 d0   83
0899 : 09 a6 cb e8 f0 dc a2 0d   a0
08a1 : d0 28 78 20 b4 e5 a2 24   f5
08a9 : dd de 08 d0 14 48 8a 0a   d3
08b1 : a8 b9 04 09 8d c1 08 b9   d3
08b9 : 03 09 8d c0 08 68 4c 00   b1
08c1 : 00 e0 0f f0 05 ca 10 e0   bc
08c9 : 30 ab a4 cb b9 81 eb cd   64
08d1 : e1 03 d0 04 a9 ff d0 eb   9e
08d9 : 8d e1 03 d0 e6 5f 31 31   c2
08e1 : 32 33 34 35 36 85 86 87   19
08e9 : 88 20 14 4d 48 58 93 13   ec
08f1 : 49 53 41 45 c5 5c 0d 11   72
08f9 : 1d 91 9d 4c 52 4f 55 44   4d
0901 : 50 54 82 09 82 09 82 09   ca
0909 : 82 09 82 09 82 09 82 09   5e
0911 : 82 09 6c 0b a3 0b cb 0b   6c
0919 : e3 0b fb 0b fb 0b 1f 0d   91
0921 : 36 0d 1e 0c 5d 0c 4d 0b   68
0929 : 6c 0c 89 0c c1 0a aa 0a   aa
0931 : aa 0a 62 0b 53 0b 47 0b   9b
0939 : 41 0b 2d 0b 37 0b 51 0d   d8
0941 : 49 0d ea 0e f2 0e 8e 10   87
0949 : 74 13 0a 13 a2 00 ad de   ca
0951 : 03 8d 21 d0 ad 16 d0 29   9e
0959 : 10 f0 03 a5 fb 2c a5 fc   08
0961 : 9d 00 04 9d 00 05 9d 00   52
0969 : 06 9d 00 07 48 ad dd 03   8e
0971 : 9d 00 d8 9d 00 d9 9d 00   3d
0979 : da 9d 00 db 68 e8 d0 e0   70
0981 : 60 ac 8d 02 a9 3f 85 3c   fe
0989 : bd ea 09 85 8c 48 38 e9   6e
0991 : 1f 8d e0 03 a9 00 8d df   a0
0999 : 03 68 c0 04 f0 2f 90 0e   68
09a1 : ca 30 1e 85 3c a9 01 8d   ec
09a9 : 20 d0 a9 3f 85 8c b9 f2   0d
09b1 : 09 8d c7 09 a9 c5 8d 0d   ad
09b9 : 0a a9 09 8d 0e 0a 20 fa   34
09c1 : 09 4c 76 08 b1 8b 91 3b   c3
09c9 : 91 3b 4c 0f 0a ca 30 f1   88
09d1 : a9 dd 8d 0d 0a a9 09 8d   9b
09d9 : 0e 0a d0 e2 b1 8b 48 b1   79
09e1 : 3b 91 8b 68 91 3b 4c 0f   17
09e9 : 0a 1f 3f 5f 7f 9f bf df   f2
09f1 : ff 64 11 51 31 65 44 04   e8
09f9 : d4 78 a9 34 85 01 38 a5   87
0a01 : 3c e9 20 85 40 a0 00 84   fd
0a09 : 8b 84 3b 4c 00 00 88 d0   f3
0a11 : fa c6 8c a5 40 c7 3c d0   1b
0a19 : f2 a9 37 85 01 58 60 a9   06
0a21 : ff cf df 03 d0 03 ce e0   82
0a29 : 03 a9 1f 85 60 a2 f8 86   85
0a31 : 5f a2 3f 86 5b 86 59 a2   17
0a39 : 47 86 5a e8 86 58 20 bf   a2
0a41 : a3 a9 00 8d f8 1f 4c 76   11
0a49 : 08 ee df 03 d0 03 ee e0   c3
0a51 : 03 a9 1f 85 60 85 59 a2   7e
0a59 : f8 86 58 e8 86 5f a2 3f   34
0a61 : 86 5b a2 48 86 5a 20 71   e5
0a69 : 0a a9 00 8d 47 3f f0 d6   d9
0a71 : 38 a5 5a e5 5f 85 fd 49   7c
0a79 : ff a8 a5 5b e5 60 aa e8   7f
0a81 : 18 a5 5f 65 fd 85 5f a5   c5
0a89 : 60 e9 00 85 60 18 a5 58   9d
0a91 : 65 fd 85 58 a5 59 e9 00   2e
0a99 : 85 59 b1 5f 91 58 c8 d0   c4
0aa1 : f9 e6 60 e6 59 ca d0 f2   17
0aa9 : 60 20 14 0b a0 27 84 8b   ec
0ab1 : a0 18 84 8c 38 e9 38 48   54
0ab9 : 8a e9 1f aa 68 4c ca 0a   7d
0ac1 : 20 14 0b a0 00 84 8b 84   1e
0ac9 : 8c 18 6d df 03 8d df 03   db
0ad1 : 85 5a 85 5f 8a 6d e0 03   6e
0ad9 : 8d e0 03 85 60 18 69 20   f5
0ae1 : 85 5b a9 00 85 58 a9 20   80
0ae9 : 85 59 ad e0 03 c9 20 90   c2
0af1 : 0b 78 a9 34 85 01 20 71   ed
0af9 : 0a 4c 04 0b a9 40 85 59   f1
0b01 : 20 bf a3 a9 37 85 01 58   73
0b09 : a6 8c a4 8b 18 20 f0 ff   d6
0b11 : 4c 50 0b a5 d1 18 65 d3   18
0b19 : aa a5 d2 e9 03 85 fe 8a   f5
0b21 : 0a 26 fe 0a 26 fe 0a 26   0e
0b29 : fe a6 fe 60 ae 8d 02 e0   67
0b31 : 02 90 19 4c 4a 0a ae 8d   16
0b39 : 02 e0 02 90 0f 4c 20 0a   26
0b41 : a6 d3 e0 27 d0 06 a6 d6   73
0b49 : e0 18 f0 03 20 d2 ff 4c   03
0b51 : 76 08 a6 d6 e0 18 d0 f4   4c
0b59 : a9 91 20 d2 ff a9 0d d0   50
0b61 : eb a0 27 a2 18 18 20 f0   5f
0b69 : ff d0 e4 a9 e8 ac 8d 02   6d
0b71 : f0 02 a9 ca 8d 92 0b ad   1b
0b79 : 16 d0 29 10 f0 03 a9 fb   09
0b81 : 2c a9 fc 8d 8e 0b 8d 98   1b
0b89 : 0b 8d 9c 0b a5 01 29 0f   08
0b91 : aa 9a 8a 09 f0 aa a5 01   49
0b99 : 09 0f 87 00 20 4d 09 4c   35
0ba1 : 76 08 a9 10 ac 8d 02 f0   a9
0ba9 : 02 a9 f0 8d c5 0b ad 16   05
0bb1 : d0 29 10 f0 03 a9 fb 2c   fe
0bb9 : a9 fc 8d c2 0b 8d c7 0b   ee
0bc1 : a5 01 18 69 00 85 01 4c   e3
0bc9 : 9d 0b ad 16 d0 29 10 f0   92
0bd1 : cb ac 8d 02 d0 06 ee dd   4b
0bd9 : 03 4c 9d 0b ce dd 03 4c   4c
0be1 : 9d 0b ad 16 d0 29 10 f0   aa
0be9 : b3 ac 8d 02 d0 06 ee de   4d
0bf1 : 03 4c 9d 0b ce de 03 4c   6c
0bf9 : 9d 0b ac 8d 02 b9 16 0c   57
0c01 : a0 07 91 40 88 10 fb e0   4c
0c09 : 0d f0 05 a9 1d 4c 41 0b   54
0c11 : a9 9d 4c 4d 0b 00 ff 55   a1
0c19 : aa 3c cc 33 c3 20 30 0c   91
0c21 : a9 8b 8d 00 03 a9 e3 8d   1b
0c29 : 01 03 68 68 4c 7b e3 a9   56
0c31 : 00 85 c6 20 44 e5 ad 11   f6
0c39 : d0 29 df 8d 11 d0 a9 06   92
0c41 : 8d 21 d0 ad 16 d0 29 ef   b5
0c49 : 8d 16 d0 a9 0e 8d 20 d0   ba
0c51 : 8d 86 02 ad 18 d0 29 f7   f4
0c59 : 8d 18 d0 60 ac 8d 02 f0   54
0c61 : 07 a0 02 a2 01 4c 85 09   28
0c69 : 4c 76 08 a9 3f 85 3c a9   8c
0c71 : 80 8d 0d 0a a9 0c 8d 0e   8a
0c79 : 0a 20 fa 09 4c 76 08 b1   6f
0c81 : 3b 49 ff 91 3b 4c 0f 0a   f9
0c89 : a9 a0 85 8b a9 98 85 3b   41
0c91 : a9 3e 85 3c 85 8c [illegible] 14   ce
0c99 : 48 a0 07 b1 8b 85 fe b1   6e
0ca1 : 3b aa 0a 8a a2 08 66 fe   07
0ca9 : 2a ca d0 fa 91 3b a5 fe   53
0cb1 : 6a 91 8b 88 10 e5 68 aa   ff
0cb9 : ca f0 1e 8a 48 a5 8b 18   e4
0cc1 : 69 08 85 8b a5 8c 69 00   65
0cc9 : 85 8c a5 3b 38 e9 08 85   63
0cd1 : 3b a5 3c e9 00 85 3c d0   ea
0cd9 : c0 a5 3b 38 e9 a8 85 3b   b2
0ce1 : 85 8b aa a5 3c e9 00 85   a9
0ce9 : 3c 85 8c a8 8a 18 69 08   3f
0cf1 : 85 8b 98 69 00 85 8c c9   81
0cf9 : 1f d0 9b ad 16 d0 29 10   ca
0d01 : f0 23 a5 fb 48 29 f0 4a   92
0d09 : 4a 4a 4a 85 fe 68 29 0f   b2
0d11 : 0a 0a 0a 0a 05 fe 85 fb   3b
0d19 : 20 4d 09 4c 76 08 ad 16   36
0d21 : d0 29 10 f0 03 4c 76 08   24
0d29 : ad 16 d0 09 10 8d 16 d0   9e
0d31 : 20 4d 09 f0 f0 ad 16 d0   cf
0d39 : 29 10 d0 02 f0 e7 ad 16   10
0d41 : d0 29 ef 8d 16 d0 d0 e8   50
0d49 : a9 27 a2 01 a0 08 d0 0f   fa
0d51 : a2 80 ad 16 d0 29 10 f0   da
0d59 : 02 a2 40 a9 00 a0 06 85   1a
0d61 : 8b 86 8c 8c 20 d0 20 bf   6c
0d69 : 0e 20 bf 0e a5 c6 d0 05   17
0d71 : e6 c6 20 b3 ee 78 20 b4   d5
0d79 : e5 c9 9d f0 13 c9 1d f0   9e
0d81 : 3a c9 20 f0 5a c9 a0 f0   1e
0d89 : 59 c9 51 d0 d9 4c 76 08   1f
0d91 : ad 16 d0 29 10 d0 12 a5   be
0d99 : 8b d0 04 a6 8c 30 c7 06   d8
0da1 : 8c 90 c3 26 8c c6 8b 10   78
0da9 : bd a5 8b d0 06 a5 8c c9   89
0db1 : 40 b0 b3 06 8c 06 8c 90   43
0db9 : ad b0 e8 a5 8b c9 27 90   72
0dc1 : 05 a6 8c ca f0 a0 ad 16   8d
0dc9 : d0 29 10 d0 0a 46 8c 90   72
0dd1 : 95 66 8c e6 8b 10 8f 46   9d
0dd9 : 8c 66 8c 10 89 30 f2 a9   f7
0de1 : 00 2c a9 ff 85 fd ad 20   a1
0de9 : d0 29 0f c9 08 f0 5e a9   20
0df1 : 20 85 41 a9 00 85 40 a5   d2
0df9 : 41 85 3c a5 40 85 3b a6   2b
0e01 : 8b f0 17 a0 07 a5 fd 91   97
0e09 : 3b 88 10 fb 18 a9 08 65   c6
0e11 : 3b 85 3b 90 02 e6 3c ca   ce
0e19 : d0 e9 a6 8c ca 8a a2 29   f7
0e21 : a4 fd f0 04 49 ff a2 09   b2
0e29 : a0 07 8d 34 0e 8e 33 0e   75
0e31 : b1 3b 49 00 91 3b 88 10   07
0e39 : f7 18 a5 40 69 40 85 40   dd
0e41 : a5 41 69 01 85 41 c9 3f   09
0e49 : 90 ad 4c 76 08 a9 21 85   ef
0e51 : 41 a9 38 85 40 a5 40 85   63
0e59 : 3b a5 41 85 3c 38 a9 27   e2
0e61 : e5 8b aa f0 19 a0 07 a5   d2
0e69 : fd 91 3b 88 10 fb 38 a5   1c
0e71 : 3b e9 08 85 3b a5 3c e9   f9
0e79 : 00 85 3c ca d0 e7 a5 8c   a0
0e81 : ae 16 d0 e0 10 90 01 0a   28
0e89 : 85 fe 38 e9 01 05 fe a2   52
0e91 : 09 a4 fd d0 04 49 ff a2   56
0e99 : 29 a0 07 8d a5 0e 8e a4   d4
0ea1 : 0e b1 3b 49 00 91 3b 88   0a
0ea9 : 10 f7 18 a5 40 69 40 85   cb
0eb1 : 40 a5 41 69 01 85 41 c9   16
0eb9 : 40 90 9a 4c 76 08 a5 8b   c7
0ec1 : 0a 0a 0a 85 40 a9 00 69   28
0ec9 : 20 85 41 a2 19 a0 07 b1   67
0ed1 : 40 45 8c 91 40 88 10 f7   82
0ed9 : 18 a5 40 69 40 85 40 a5   7e
0ee1 : 41 69 01 85 41 ca 10 e5   3e
0ee9 : 60 a2 20 a0 0a a9 00 f0   86
0ef1 : 06 a2 3e a0 05 a9 07 85   b1
0ef9 : 8b 86 8c 8c 20 d0 20 27   d3
0f01 : 10 20 27 10 a5 c6 d0 05   cb
0f09 : e6 c6 8d 77 02 78 20 b4   72
0f11 : e5 c9 91 f0 13 c9 11 f0   03
0f19 : 3a c9 20 f0 61 c9 a0 f0   27
0f21 : 60 c9 51 d0 d9 4c 76 08   be
0f29 : a5 8c c9 20 d0 04 a5 8b   66
0f31 : f0 cc a5 8b 29 07 f0 0c   09
0f39 : c6 8b a5 8b c9 ff d0 be   fd
0f41 : c6 8c d0 ba a5 8b 38 e9   44
0f49 : 39 85 8b a5 8c e9 01 85   03
0f51 : 8c d0 ab a5 8c c9 3e 90   16
0f59 : 06 a5 8b c9 07 f0 9f a5   10
0f61 : 8b 29 07 49 07 f0 08 e6   52
0f69 : 8b d0 93 e6 8c d0 8f a5   f7
0f71 : 8b 18 69 39 85 8b a5 8c   ee
0f79 : 69 01 85 8c d0 80 a9 00   0d
0f81 : 2c a9 ff 85 3c ad 20 d0   86
0f89 : 29 0f c9 0a f0 72 a5 8c   40
0f91 : c9 3e b0 24 a5 8b 29 f8   77
0f99 : 18 69 41 85 5f a5 8c 69   8f
0fa1 : 01 85 60 a2 40 86 5a ca   08
0fa9 : 86 58 a9 3f 85 5b 85 59   aa
0fb1 : a5 3c 8d 3f 3f 20 bf a3   fb
0fb9 : a9 28 85 40 a5 8b a2 07   2f
0fc1 : 87 3b 29 f8 85 8b ad 20   fb
0fc9 : d0 29 0f aa a5 3c a0 00   06
0fd1 : c4 3b f0 09 e0 05 f0 02   8e
0fd9 : 91 8b c8 d0 f3 91 8b c8   07
0fe1 : c0 08 b0 09 e0 0a f0 02   19
0fe9 : 91 8b c8 d0 f3 a5 8b 69   f9
0ff1 : 07 85 8b a5 8c 69 00 85   71
0ff9 : 8c c6 40 d0 cf 4c 76 08   5c
1001 : a5 8c c9 21 90 b2 85 5b   ee
1009 : a5 8b 29 f8 85 5a a9 20   ef
1011 : 85 60 85 59 a2 01 86 58   50
1019 : ca 86 5f a5 3c 8d ff 1f   21
1021 : 20 71 0a 4c b9 0f a5 8b   c8
1029 : 85 40 a5 8c 85 41 a0 28   fe
1031 : a2 00 a1 40 49 ff 81 40   5f
1039 : 18 a5 40 69 08 85 40 90   30
1041 : 02 e6 41 88 d0 ec 60 09   20
1049 : 0e 20 c8 41 52 44 43 4f   b5
1051 : 50 59 2d cd 41 4b 45 52   7b
1059 : 20 20 20 57 52 49 54 54   e6
1061 : 45 4e 20 42 59 20 c3 48   54
1069 : 52 2e 20 cb 55 52 54 53   34
1071 : 20 20 20 20 20 20 20 20   71
1079 : 20 20 2d 20 c4 49 53 4b   73
1081 : 57 4f 52 4b 2d cd 4f 44   85
1089 : 55 53 0d 0d 00 20 30 0c   47
1091 : a0 00 b9 48 10 f0 06 20   8a
1099 : d2 ff c8 d0 f5 20 cc ff   4a
10a1 : 20 60 a5 86 7a 84 7b a2   2a
10a9 : 07 20 73 00 c9 00 f0 f0   df
10b1 : dd d0 10 f0 08 ca 10 f8   22
10b9 : a2 11 4c e8 10 8a 0a aa   e7
10c1 : bd d8 10 8d ce 10 bd d9   b8
10c9 : 10 8d cf 10 4c 00 00 4e   f7
10d1 : 24 40 51 58 5f 5c 3f 4c   e3
10d9 : 12 2a 11 c2 11 54 08 21   b3
10e1 : 0c b5 12 70 12 55 12 8a   84
10e9 : 0a aa 20 f4 10 20 7a a6   28
10f1 : 4c a1 10 bd 26 a3 85 22   a3
10f9 : bd 27 a3 85 23 a9 00 85   6e
1101 : 13 20 cc ff 20 3f ab a0   43
1109 : 00 b1 22 48 29 7f 20 d2   28
```

Listing »Hardmaker«. Bitte mit dem MSE eingeben

```
1111 : ff c8 68 10 f4 a0 00 b9   58
1119 : 6a a3 20 d2 ff c8 c0 06   0c
1121 : 90 f5 20 44 12 20 44 12   94
1129 : 60 20 2a 12 90 06 20 44   a8
1131 : 12 4c a1 10 20 44 12 a5   8b
1139 : 7b 85 bc a5 7a 85 bb 20   5e
1141 : 64 12 84 b7 ad e2 03 85   cf
1149 : ba a9 60 85 b9 20 d5 f3   7c
1151 : a5 ba 20 b4 ff a5 b9 20   46
1159 : 96 ff a9 00 85 90 a0 03   bf
1161 : 84 40 20 a5 ff 85 41 a4   3d
1169 : 90 d0 41 20 a5 ff a4 90   c4
1171 : d0 3a a4 40 88 d0 e9 48   d7
1179 : a0 06 20 3f ab 88 d0 fa   45
1181 : 68 a6 41 20 cd bd 20 3f   5a
1189 : ab 20 3f ab 20 a5 ff a6   06
1191 : 90 d0 19 aa f0 06 20 d2   8a
1199 : ff 4c 8d 11 a5 91 10 0c   83
11a1 : ad 8d 02 d0 fb 20 44 12   a5
11a9 : a0 02 d0 b4 20 44 12 a5   cd
11b1 : 90 29 83 d0 06 20 42 f6   29
11b9 : 4c a1 10 20 e0 11 4c a1   e9
11c1 : 10 20 2a 12 b0 0a 20 73   71
11c9 : 00 c9 00 d0 06 20 e0 11   cf
11d1 : 4c a1 10 c9 24 d0 03 4c   99
11d9 : 2a 11 20 09 12 90 f1 a0   64
11e1 : 00 84 90 ad e2 03 85 ba   cf
11e9 : 20 b4 ff a9 6f 85 b9 20   e3
11f1 : 96 ff 20 a5 ff 24 90 70   88
11f9 : 05 20 d2 ff d0 f4 20 ab   4f
1201 : ff 20 44 12 20 44 12 60   91
1209 : ad e2 03 85 ba 20 b1 ff   0c
1211 : a9 6f 85 b9 20 93 ff 20   e9
1219 : 79 00 20 a8 ff 20 73 00   7e
1221 : c9 00 d0 f6 20 ae ff 18   a5
1229 : 60 ad e2 03 85 ba a2 00   32
1231 : 86 90 20 b4 ff 20 ab ff   4e
1239 : a5 90 f0 eb a2 0a 20 f4   c5
1241 : 10 38 60 48 a9 0d 20 d2   b8
1249 : ff 68 60 20 9b b7 8e e2   10
1251 : 03 4c a1 10 20 79 a5 20   89
1259 : 73 00 20 73 00 20 a0 aa   1c
1261 : 4c a1 10 a0 00 b1 7a f0   ef
1269 : 05 c8 c0 28 90 f7 60 20   92
1271 : 2a 12 b0 34 a9 20 8d e0   eb
1279 : 03 a9 00 8d df 03 20 73   80
1281 : 00 a5 7b 85 bc a5 7a 85   d1
1289 : bb 20 64 12 84 b7 ad e2   32
1291 : 03 85 ba a9 00 85 b9 a2   93
1299 : 00 86 0a a0 20 20 d5 ff   cd
12a1 : b0 09 20 44 12 20 e0 11   2e
12a9 : 4c a1 10 20 e8 12 20 e0   2f
12b1 : 11 4c a1 10 20 2a 12 b0   50
12b9 : ef 20 73 00 a5 7b 85 bc   5b
12c1 : a5 7a 85 bb 20 64 12 84   f3
12c9 : b7 a9 00 ae e2 03 a8 20   54
12d1 : ba ff a9 00 85 69 a9 20   80
12d9 : 85 6a a2 40 a0 3f a9 69   c1
12e1 : 20 d8 ff 90 bd b0 c4 a2   39
12e9 : 00 bd f6 12 f0 06 20 d2   2d
12f1 : ff e8 d0 f5 60 0d 4f 50   a3
12f9 : 45 52 41 54 49 4f 4e 20   cb
1301 : 53 54 4f 50 50 45 44 0d   b7
1309 : 00 ad 16 d0 29 10 f0 11   78
1311 : a9 3f 85 3c a9 25 8d 0d   57
1319 : 0a a9 13 8d 0e 0a 20 fa   16
1321 : 09 4c 3f 0d a2 03 b1 3b   41
1329 : 3d 68 13 f0 34 48 3d 6c   d1
1331 : 13 f0 13 68 3d 70 13 d0   d3
1339 : 28 98 4a b0 24 b1 3b 5d   cd
1341 : 68 13 91 3b d0 1b 68 84   8f
1349 : fd 8a 45 fd 4a b0 09 bd   66
1351 : 68 13 11 3b 91 3b d0 09   37
1359 : bd 68 13 49 ff 31 3b 91   d2
1361 : 3b ca 10 c2 4c 0f 0a c0   45
1369 : 30 0c 03 40 10 04 01 80   8e
1371 : 20 08 02 a9 00 8d 20 d0   da
1379 : 20 a0 13 4c 76 08 ea ea   61
1381 : ea ea ea ea ea ea ea ea   80
1389 : ea ea ea ea ea ea ea ea   88
1391 : ea ea ea ea ea ea 56 f2   4d
1399 : df 2d 00 52 d3 eb e7 a9   e9
13a1 : 20 8d f5 14 a9 04 85 ba   ce
13a9 : a2 00 86 90 86 fe 20 b1   43
13b1 : ff 20 ae ff a6 90 f0 01   21
13b9 : 60 86 b9 86 b7 e8 86 b8   ea
13c1 : 20 c0 ff a6 b8 20 c9 ff   ca
13c9 : a9 ff 85 61 a9 07 8d f6   f6
13d1 : 14 a9 1c 85 97 a9 00 8d   53
13d9 : f1 14 a9 28 8d f3 14 a2   52
13e1 : 04 bd ce 14 20 d2 ff ca   28
13e9 : 10 f7 a9 00 85 63 85 64   b2
13f1 : ad f1 14 85 65 a9 00 8d   0b
13f9 : f7 14 a5 63 a6 64 a4 65   bb
1401 : 20 93 14 ae f7 14 a5 ad   d8
1409 : a0 00 b1 ac ae f7 14 9d   e1
1411 : f8 14 e6 65 e8 8e f7 14   85
1419 : ec f6 14 d0 dd a9 00 a0   0c
1421 : 07 d0 02 d0 b5 ae f6 14   00
1429 : 1e f8 14 2a ca 10 f9 25   6d
1431 : 61 09 80 20 d2 ff ad 8d   3a
1439 : 02 29 01 d0 f9 a5 91 10   5d
1441 : 3d 88 10 e1 a5 63 18 69   ab
1449 : 08 85 63 90 02 e6 64 ce   85
1451 : f3 14 d0 9c a9 0d 20 d2   3f
1459 : ff ad f1 14 18 69 07 8d   32
1461 : f1 14 c6 97 f0 02 d0 bb   db
1469 : a9 04 cd f6 14 f0 0f 8d   87
1471 : f6 14 a9 01 85 97 a9 0f   d6
1479 : 85 61 d0 ea a9 01 85 fe   f7
1481 : a9 0f 20 d2 ff a9 0d 20   d6
1489 : d2 ff 20 cc ff a9 01 4c   e6
1491 : c3 ff 85 14 86 15 98 4a   40
1499 : 4a 4a aa bd d3 14 85 ad   ba
14a1 : 8a 29 03 aa bd ed 14 85   7d
14a9 : ac 98 29 07 18 65 ac 85   37
14b1 : ac a5 14 29 f8 85 63 ad   ff
14b9 : f5 14 18 65 ad 85 ad a5   74
14c1 : ac 18 65 63 85 ac a5 ad   ef
14c9 : 65 15 85 ad 60 50 00 10   79
14d1 : 1b 08 00 01 02 03 05 06   69
14d9 : 07 08 0a 0b 0c 0d 0f 10   4e
14e1 : 11 12 14 15 16 17 19 1a   56
14e9 : 1b 1c 1e 1f 00 40 80 c0   03
14f1 : 00 00 00 00 00 00 00 00   f2
14f9 : 00 00 00 00 00 00 00 80   fb
```

Listing »Hardmaker« (Schluß)

So testen wir — das sind unsere Referenzdrucker

Fortsetzung von Seite 23

Preisklasse I (bis 1000 Mark)

In der Ausgabe 2/86 konnten wir dem Citizen 120 D (Bild 2) als erstem Drucker dieser Preisklasse das Prädikat »Referenzdrucker« verleihen. Durch seine gemessen am relativ niedrigen Preis von 998 Mark erstaunlichen Leistungen, zu denen eine NLQ-Schrift, eine Druckgeschwindigkeit von 140 Zeichen pro Sekunde in der Normalschrift und 24 Zeichen pro Sekunde in der NLQ-Schrift sowie einige praktische Detaillösungen gehören, konnte er den Titel erringen. In dieser Preisklasse muß man natürlich auch Abstriche machen: So ist der 120 D nicht so massiv aufgebaut wie die Drucker der höheren Preisklassen, aber er erfüllt seine Aufgabe bestens und hat mit seiner zweijährigen Garantie in diesem Bereich neue Maßstäbe gesetzt.

Preisklasse II (bis 1400 Mark)

Er ist brandneu und er ist gut — der Star NL-10 (Bild 3). Einen ausführlichen Testbericht finden Sie in dieser Ausgabe. Dort steht auch, warum der NL-10 so ideal zu Commodore-Computern paßt. Er hat in dieser Preisklasse den Star SG-10 (sein Vorgängermodell) abgelöst, der sich mittlerweile bei vielen C 64-Besitzern großer Beliebtheit erfreut. Der NL-10 (1 145 Mark) ist wohl der beste Schwarzweiß-Drucker, den man in dieser Preislage derzeit erhalten kann.

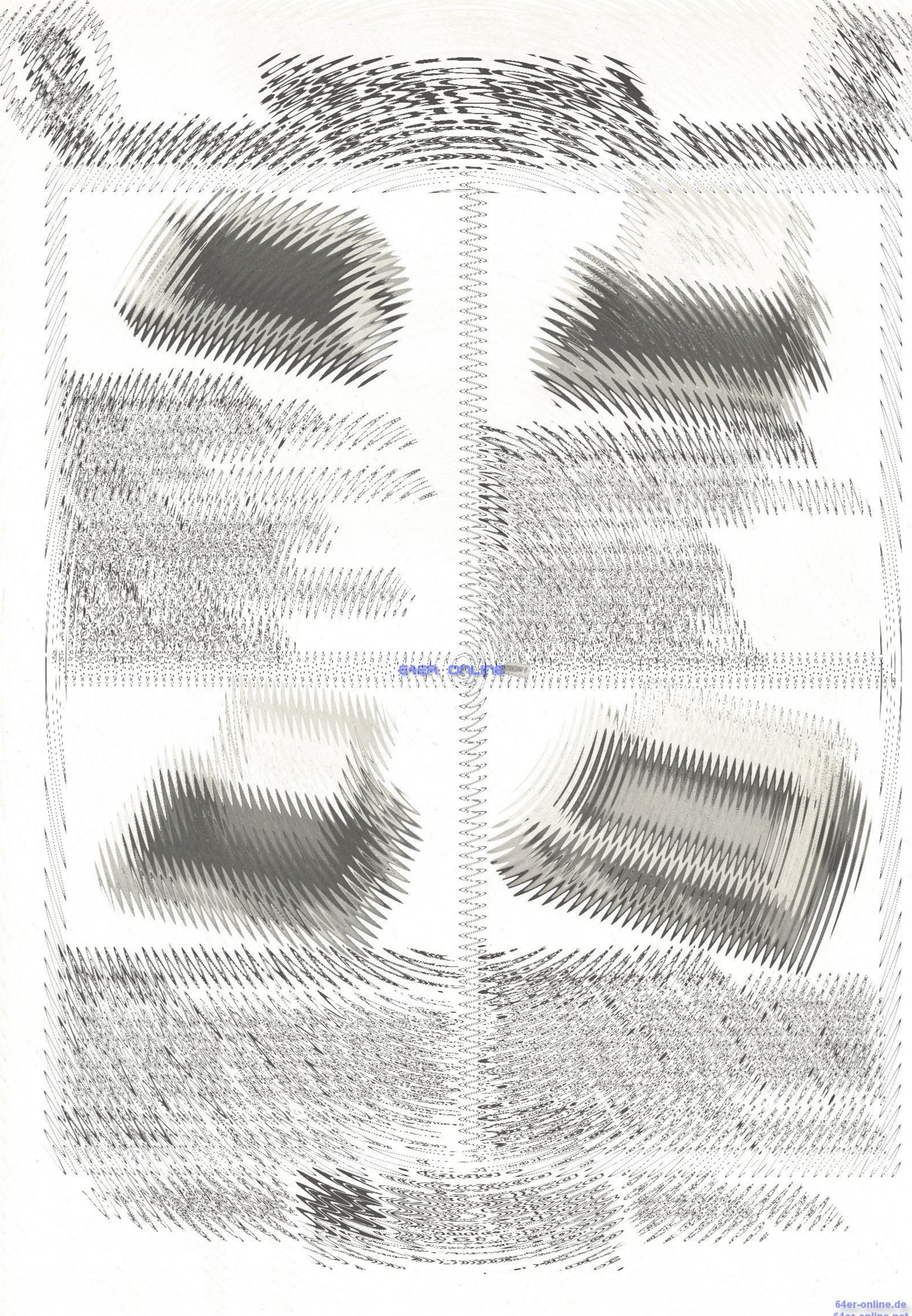
Preisklasse III (bis 2000 Mark)

Jetzt kommt Farbe ins Spiel. In der höchsten Preisklasse konnte sich der Fujitsu DX 2100 (Bild 4) gegenüber dem Epson FX-85, der diesen Posten ein halbes Jahr bestens bekleidet hat, erfolgreich durchsetzen. Das für Personal- und Heimcomputer ideale Konzept des DX 2100 mit extrem schnellem (220 Zeichen) und schönem Druck (NLQ) und einer sehr guten Farbfähigkeit macht ihn zum Spitzenreiter unserer Referenzdrucker. Mit einem Preis von 1932 Mark und 456 Mark Aufpreis für die Farboption ist er zwar nicht billig, aber gemessen an seinen Leistungen preiswert.

Was bringt die Zukunft

Der Druckermarkt ist einer der bewegtesten und am heißesten umkämpften in der Computerbranche überhaupt. Ein wildes Pulsieren von neuen Modellen und ständiger Leistungssteigerung bei sinkenden Preisen machen es für den Kunden zum interessanten Abenteuer, dieser Entwicklung zuzusehen. Das Jahr 1986 wird sicherlich voller Überraschungen sein, denn für die Druckerhersteller heißt es heute die Marktanteile von morgen zu sichern. Dazu lassen sie sich einiges einfallen. Einen Drucker wie beispielsweise den NL-10 oder den DX-2100 hätte man noch vor drei Jahren nicht unter vier- bis fünftausend Mark erhalten können, wenn überhaupt. Aber der Entwicklung sind Grenzen gesetzt, denn Drucker bestehen zum großen Teil aus Mechanik, und die ist in der Produktion teuer. Die Druckerhersteller sind wohl mittlerweile an dem Punkt angelangt, an dem sich durch Kostensenkungen bei der Elektronik keine wesentlichen Preissenkungen beim Endgerät mehr erzielen lassen. Soll die mechanische Qualität erhalten bleiben, so dürften sich die Preise in Zukunft wohl nur noch wenige Hunderter unter dem derzeitigen Niveau einpendeln. Ein zusätzliches Verkaufsargument wird dann nur noch durch zusätzliche Leistungssteigerung zu erreichen sein — man darf sich freuen! (aw)


64ER ONLINE





Fehlerteufelchen

Roulette 128, Sonderheft 1/86, Seite 62ff

In Zeile 1103 steht im veröffentlichten Listing »... THEN 1130«. Die Zeile 1130 existiert jedoch nicht. Es muß logischerweise heißen »... THEN 2000«. In manchen Heften sind zwei Zeilen nicht optimal zu lesen. Deshalb hier noch einmal diese beiden Zeilen:

```
2347 IF SE=20 OR K(S)=0 THEN 2380
2350 GOSUB 6000.
```

Simons Basic und Turbo Tape, Ausgabe 7/85, Seite 85.

In der zweiten Zeile des Umschreibeprogramms befindet sich ein Fehler. Statt »IF C>19« sollte es heißen »IF C>191«.

Probleme mit C 128, Ausgabe 2/86, Seite 17.

Auf dieser Seite erklärt Willi Brechtel, wie sich aus einem Programm heraus auf die DIN-Tastatur umschalten läßt. Das funktioniert jedoch nicht mit der Befehlsfolge »POKE 0, PEEK(0) AND 64: POKE 1,0« sondern mit »POKE 0, PEEK (0) OR 64: POKE 1,0«.

Spline — das computergesteuerte Kurvenlineal, Sonderheft 2/86, Seite 96ff.

Durch eine fehlende DIM-Anweisung kommt es bei der Eingabe größerer Datenmengen zu einer Fehlermeldung. Die Programmzeile 670 muß wie folgt ergänzt werden:

```
670 DIM X (AK,N1), X$(AK,N1), Y(AK,N1), SW(AK,N1).
```

Von Basic zu Assembler (2), Ausgabe 2/86, Seite 154.

Im dokumentierten SMON-Listing wurde eine Zeile vergessen: »4013 INC $FB«.

Nicht nur ein Geheimdienst: CIA, Ausgabe 2/86, Seite 94.

In der rechten Spalte steht: Zur Einstellung der Minuten ist anzugeben: POKE 56586,39 (45 Minuten). Das ist falsch. Der richtige Befehl lautet: POKE 56586,69.

Rechnungshelfer, Sonderheft 7/85, Seite 80.

Die abgedruckte Zeile 1390 ist falsch. Richtig lautet sie:

```
1390 IF SL$=''K'' THEN Z=Z+1: INPUT ''KOMMENTAR''; C$ (Z,3): GOTO1320
```

Wie wär's mit:..., Sonderheft 2/86, Seite 80

Im Assemblerprogramm auf dieser Seite haben sich einige Fehler eingeschlichen. Folgende Zeilen sind zu ergänzen:

```
45- SEI
355- CLI
```

Folgende Zeilen sind abzuändern:

```
152- AND #251
210- WE INC $FC
230- INC $FD
330- LDA #$37
```

Die Zeile 340 ist ersatzlos zu streichen.

Streifzüge durch die Grafikwelt (Teil 2), Ausgabe 11/85, Seite 149ff.

Auf Seite 154, Listing 9: Das Programm zeichnet in dieser Form das Achsenkreuz falsch ein. Folgende Zeilen reparieren den Schaden:

```
336      GOTO 375
356      RETURN
391      ZZ=FNZ (-ZU):XT=XU-
         ZZ:XH=XO-ZZ:YT=YU-
         ZZ:YH=YO-
         ZZ:TRS,XT,XH,YT,YH: GO-
         SUB 355
```

Man muß darauf achten, daß »ZU« kleiner und »ZO« größer als Null gewählt wird.

Sprites und Shapes auf dem C128, Sonderheft 1/86, Seite 32ff

In diesem Artikel wurde das Listing »24 SPRITES« vergessen. Das holen wir jetzt nach (s. List.).

Assembler ist keine Alchimie, Sonderheft 8/85, Seite 4ff

Der lange Assemblerkurs war anscheinend ein bevorzugtes Angriffsziel des Fehlerteufels:

Seite 5, Kapitel 2, 2. Absatz: Im Basic-Programm sind für den VC 20 in der Grundversion anstelle von 54272 einzusetzen 30720. Außerdem wird die Zeile 30 im Listing darunter für den VC 20 in der Grundversion so geschrieben:

```
30 DATA160,255,162,6,169,1,153,
0,30,138,153,0,150,136,208,244,
96
```

Seite 10, Kapitel 9, linke Spalte unten: Die Angaben über den Farbspeicher des VC 20 in der Grund- und in der erweiterten Version sind vertauscht. Grundversion: $9600, erweiterte Version $9400. Ebenso vertauscht sind in diesem Kapitel im letzten Satz die Angaben für die beiden Versionen. Dort muß es heißen: Grundversion 30 und 150, erweiterte Version 16 und 148.

Seite 11, Kapitel 10, letzter Satz muß lauten: Ein vergleichbares Basic-Programm braucht dazu etwa tausendmal so lange: zirka 0,05 Sekunden.

Seite 16, letzter Absatz vor Bild 8: Die Zahl -32768 kann auch als A% = -32768 oder in B% = -32765:B% = B% + -3 auftreten.

Seite 18, rechts, drittes Rechenbeispiel: Wir addieren zwei negative Zahlen, -4 und -2:

```
-4      1111 1100
-2      1111 1110
+       +
-6(1)   1111 1010
```

Seite 19, Seite 21 und Seite 34: In einigen Assemblerprogrammen sind Bezeichnungen verrutscht. So muß es beispielsweise auf Seite 19 statt

```
120     BLDA 1301 und so weiter
        heißen
120B    LDA 1301
```

Seite 31 und 32. Hier sind unkontrollierte Minuszeichen in die Exponenten gerutscht.

Anstelle von

10↑-2 = 1/10↑-2 = 0.01
3.14 * 10↑-2 = 0.314 * 10↑-3
2.69 * 10↑-19 Heliumatome
6.02E+23 = 6.02 * 10↑-23
+1.70141183 * 10↑-38

muß es heißen

10↑-2 = 1/10↑2 = 0.01
3.14 * 10↑2 = 0.314 * 10↑3
2.69 * 10↑19 Heliumatome
6,02E+23 = 6.02 * 10↑23
+1.70141183 *10+38

Seite 34, links in der Mitte muß es statt »Das X-Register verwenden wir als Index und laden es mit dezimal 40 = $27« lauten: Das X-Register verwenden wir als Index und laden es mit dezimal 39 = $27.

Profi-Auflösung für den MPS 801, Ausgabe 2/86, Seite 61

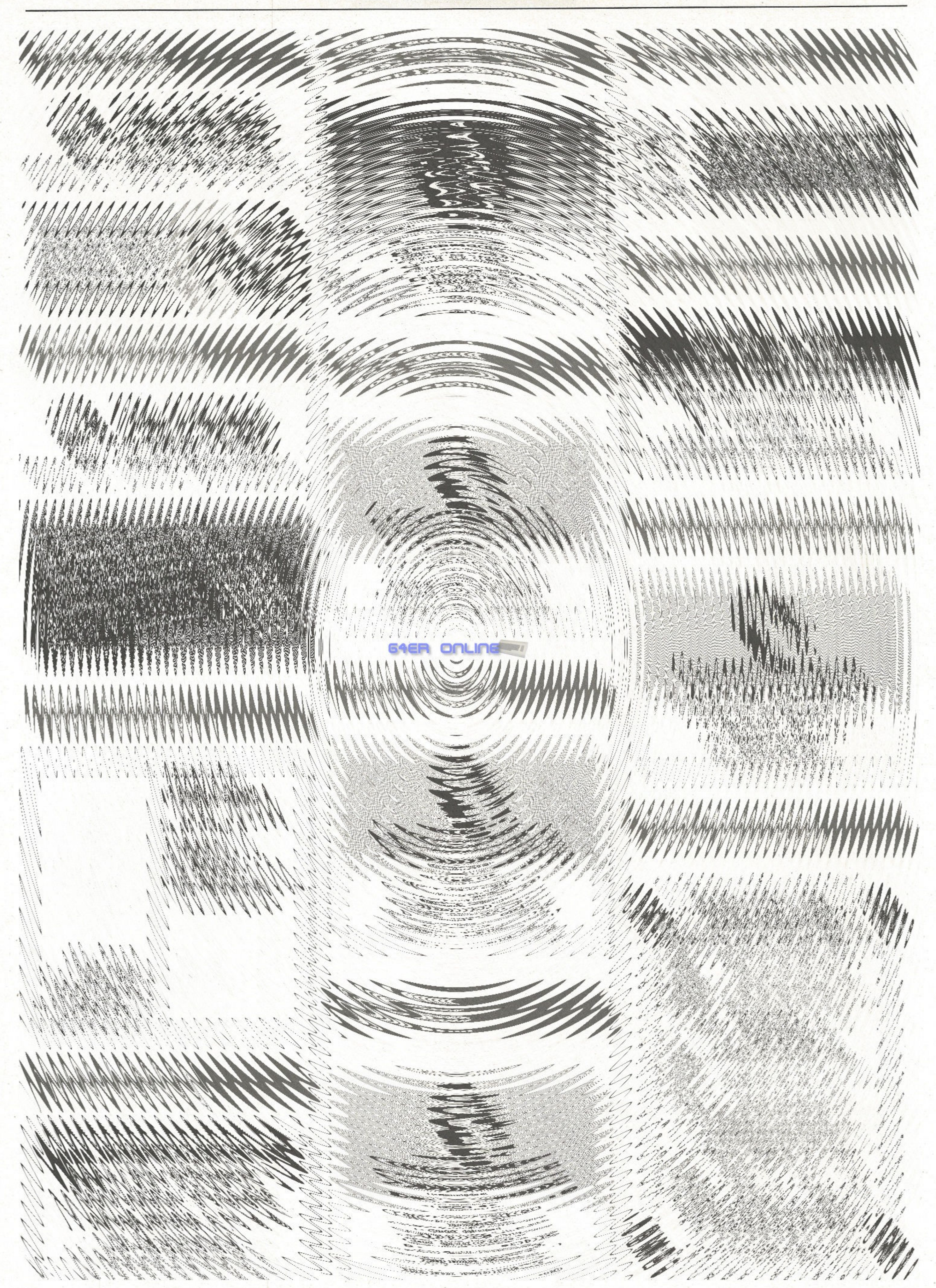
Im abgedruckten MSE-Listing 1 stimmt die erste Programmzeile nicht. Um diesen Druckfehler zu beseitigen, ist das Programm »MPS-Support« mit dem MSE zu laden. Anschließend ist die Tastenkombination »CTRL-N« erforderlich. Als Startadresse geben Sie »0801« ein. Nun läßt sich die korrekte Zeile eintippen. Sie lautet:

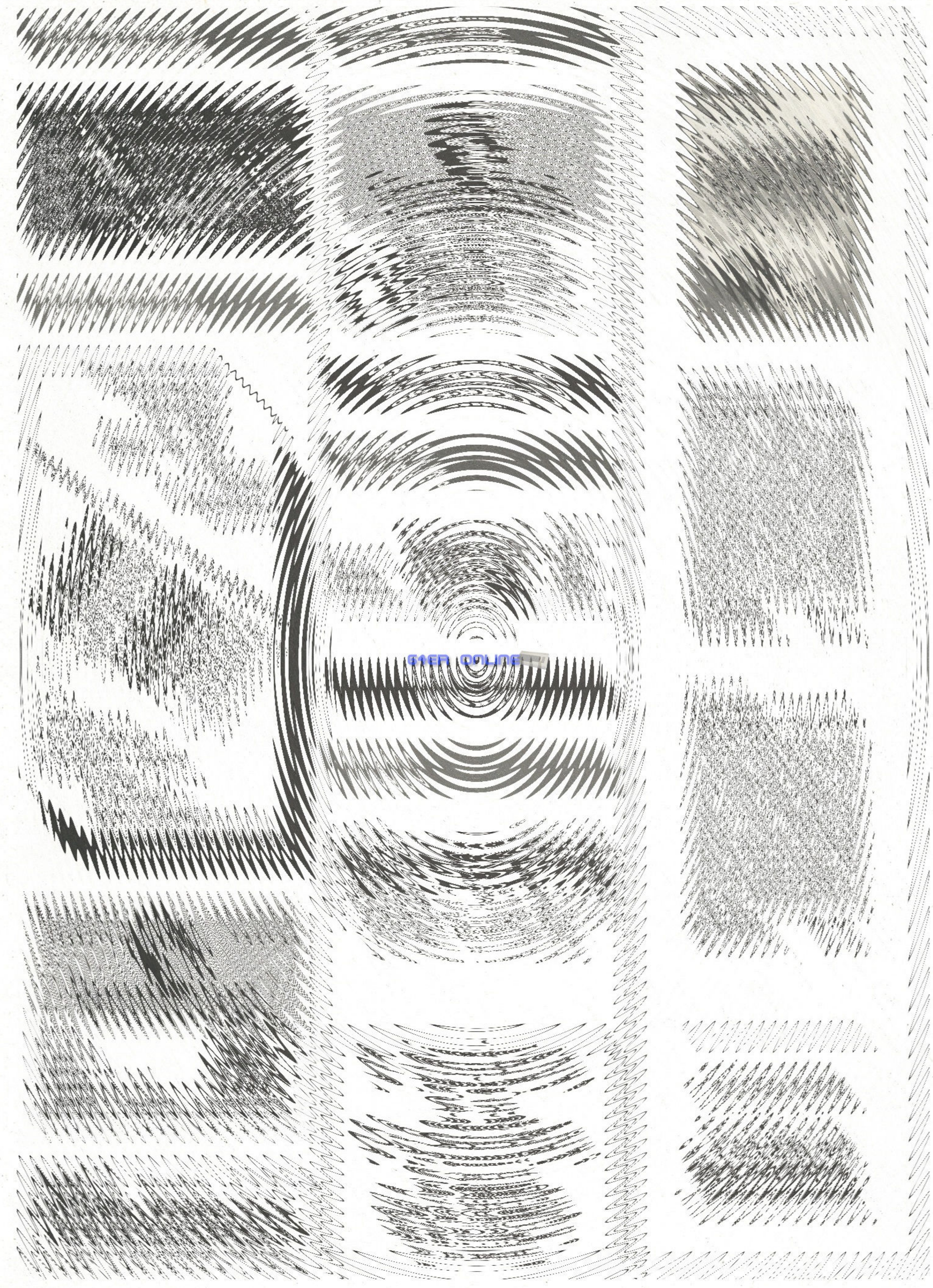
```
0801:0d 08 00 00 9e 32 30 36 bb
```

Ist die Zeile akzeptiert worden, können Sie die verbesserte Version durch Drücken der RUN/STOP-Taste gefolgt von der Kombination »CTRL-S« speichern.

```
10 REM ***** 24 SPRITES *****
20 I=0: J=1: K=2: L=3: M=4: N=5: O=6: P=7:
   B$="": DIM A$(24)
30 BLOAD "SCENE1"
40 FOR I=1 TO 8: SPRSAV I,A$(I): NEXT I
50 BLOAD "SCENE2"
60 FOR I=1 TO 8: SPRSAV I,A$(I+8): NEXT I
70 BLOAD "SCENE3"
80 FOR I=1 TO 8: SPRSAV I,A$(I+16): NEXT I
90 PRINT CHR$(147)
100 SPRCOLOR 3,6: COLOR 0,1: COLOR 4,1: COL
    OR 5,8
110 MOVSPR 1,62,86: MOVSPR 2,111,86: MOVSPR
     3,160,86: MOVSPR 4,209,86: MOVSPR 5,62
    ,129
120 MOVSPR 6,111,129: MOVSPR 7,160,129: MOV
    SPR 8,209,129
130 SPRITE 1,1,8,1,1,1,1: SPRITE 2,1,14,1,1
    ,1,1: SPRITE 3,1,2,1,1,1,1
140 SPRITE 4,1,11,1,1,1,1: SPRITE 5,1,13,1,
    1,1,1: SPRITE 6,1,5,1,1,1,1
150 SPRITE 7,1,1,1,1,1,1: SPRITE 8,1,16,1,1
    ,1,1
160 I=0
170 DO
180 : I=I+1: IF I>24 THEN I=1: SPRSAV A$(I)
    ,1: ELSE SPRSAV A$(I),1
190 : J=J+1: IF J>24 THEN J=1: SPRSAV A$(J)
    ,2: ELSE SPRSAV A$(J),2
200 : K=K+1: IF K>24 THEN K=1: SPRSAV A$(K)
    ,3: ELSE SPRSAV A$(K),3
210 : L=L+1: IF L>24 THEN L=1: SPRSAV A$(L)
    ,4: ELSE SPRSAV A$(L),4
220 : M=M+1: IF M>24 THEN M=1: SPRSAV A$(M)
    ,5: ELSE SPRSAV A$(M),5
230 : N=N+1: IF N>24 THEN N=1: SPRSAV A$(N)
    ,6: ELSE SPRSAV A$(N),6
240 : O=O+1: IF O>24 THEN O=1: SPRSAV A$(O)
    ,7: ELSE SPRSAV A$(O),7
250 : P=P+1: IF P>24 THEN P=1: SPRSAV A$(P)
    ,8: ELSE SPRSAV A$(P),8
260 : GET B$: IF B$="_" THEN EXIT
270 LOOP
280 END
```

Listing »24 SPRITES«


64ER ONLINE



Tips & Tricks für Einsteiger

Haben Sie Ladeprobleme mit fremden Kassetten? Oder möchten Sie auf einfachste Weise den Joystick abfragen? Auch diesmal haben wir wieder ein paar interessante Hinweise für diejenigen Leser, die sich auf dem C 64 »noch nicht so ganz fit« fühlen.

Man kann es nur immer wieder betonen: Wir sind eine Zeitschrift, die von ihren Lesern lebt! Helfen Sie den Einsteigern unter den C 64-Fans durch einen besonderen POKE, eine kurze, aber sehr nützliche Basic-Routine oder einfach durch eine Problemlösung, die Sie als besonders genial erachten. Schreiben Sie uns unter dem Stichwort »Tips & Tricks«!

Datasette richtig justiert

Jetzt ist Schluß mit »LOAD ERROR« beim Abspielen von fremden Kassetten.

Jeder, der eine Datasette besitzt, weiß, daß diese bei einem anderen C 64-Besitzer meist nicht mit derselben Tonkopfeinstellung justiert ist. Bekommt man eine heißerwartete Kassette mit Spielen oder anderen Programmen, womöglich noch im Turbo-Tape-Format, ist die Enttäuschung groß, wenn man feststellt, daß die Tonkopfjustierung von Aufnahme- und Wiedergabegerät abweicht.

»SYNCHRO JUSTAGE« (Listing 1) brennt einer Kassette ein 50-Herz-Synchronsignal auf, das nach der Übergabe der Kassette mit »SYNCHRO JUSTAGE« auf dem Bildschirm ausgewertet wird. Der Empfänger stellt nun mit der Einstellschraube seinen Tonkopf so ein, daß das Flackern des Bildschirmes minimal wird. Je weniger er flackert, desto besser hat man justiert. Wandert der Synchronbalken nach oben, läuft die Datasette bei der Wiedergabe schneller als bei der Aufnahme und umgekehrt. Dann gilt es zusätzlich die Motorgeschwindigkeit zu überprüfen und optimal einzustellen, bis der schwarze Balken stillsteht.

Ich tausche nur noch Kassetten mit Synchron-Signal aus und kann keine Probleme mit meiner Datasette beklagen. Man muß zwar von Zeit zu Zeit seine Tonkopfeinstellung einer anderen anpassen, kann aber seinem Bekannten sagen, wieviele Umdrehungen seine Einstellschraube von der eigenen abweicht. (Reinhard Abdel-Hamid/tr)

Betrifft: Joystick

Jeder Neuling unter den C 64-Fans wird sich früher oder später fragen, warum Commodore gleich zwei Joystick-Ports einbaute, es aber versäumte, das Basic des C 64 um eine Funktion zu bereichern, diese Joysticks auch abzufragen. Viele andere Computer haben einen »JOY(X)«-Befehl, der die Richtung angibt, in die der Joystick gerade gedrückt wird.

Auf dem C 64 wurden solche Abfragen bisher mit langwierigen IF-THEN-Sequenzen über die PEEK-Funktion realisiert. Es gibt aber einen sehr viel eleganteren Weg: die DEF FN-Funktion.

Wenn man nämlich am Anfang eines Basic-Programms definiert:

```
DEFFNJOY(X)=INT((LOG(255.5-(PEEK(56322-X)OR224)))/
LOG(2)+2)
```

so läßt sich über »`PRINT FN JOY(x)`« der Joystick abfragen. »x« gibt dabei an, ob man die Position von Joystick-Port 1 oder 2 wissen möchte. Es entsprechen: 1: Nullstellung, 2: oben, 3: unten, 4: links, 5: rechts und 6: Feuerknopf

Über eine »ON FN JOY(x) GOTO ...«-Anweisung ließe sich dann äußerst schnell in die entsprechenden Unterprogramme verzweigen. (Henning Zipf/tr)

Das elektronische Tagebuch

Uralt schon ist das Bestreben vieler Leute, ihre geheimsten und privatesten Aufzeichnungen so gut wie möglich vor allzu neugierigen Mitmenschen zu schützen. Und wer kennt sie nicht, die Kladde mit dem Schlößchen, der solche Geheimnisse anvertraut werden — mit dem blauäugigen Glauben, es gäbe niemanden, der auf die Idee käme, sich dem Schloß mit einer aufgebogenen Büroklammer zu nähern...

Es liegt also nahe, sich nach einem wirkungsvolleren Schutz umzusehen. Der Computer bietet sich für solch verantwortungsvolle Dienste geradezu an. Es fehlt also nur noch das passende Programm.

Das »elektronische Tagebuch« in seiner Mikroausführung (Listing 2) ist da genau richtig. Die Vorteile liegen auf der Hand: Das Programm ist sehr kurz (eine Bildschirmseite) und zugleich recht komfortabel. Seine Benutzung bereitet also weder bei der Eingabe noch bei der Anwendung großen Aufwand.

Das Programm läßt sich theoretisch auf jedem Commodore-Computer verwenden, einzige Voraussetzung ist natürlich ein externes Speichermedium, sprich: Datasette oder Floppy.

Die vorliegende Version ist für den Gebrauch mit dem C 64 und Diskettenstation vorgesehen, die Anleitung zum Umschreiben auf Kassettenbetrieb und Hinweise für Benutzer anderer Computer folgen weiter unten.

Das Programm ist im Vergleich zur beschriebenen Kladde recht sicher. Absolut unknackbar ist es zwar nicht, aber es gehört Intelligenz statt Fingerspitzengefühl dazu, aus einer polyalphabetisch verschlüsselten und zu scheinbar beziehungslosen Zahlenfolgen umgewandelten Buchstabenkombination ein lesbares Satzgefüge (sprich: Text) zu machen. Außerdem steht es jedem frei, seine eigene Chiffriermethode zu entwickeln.

Doch kommen wir zum Programm selbst. Der Aufbau ist denkbar einfach: Gleich nach dem Start wird aus Sicherheitsgründen (für Ihren Text) die RUN/STOP-RESTORE-Funktion ausgeschaltet. In dem jetzt erscheinenden Mini-Menü müssen Sie sich entscheiden, ob Sie einen Text eingeben und codieren oder ob Sie ihn decodieren und lesen wollen. Dann werden Sie daran erinnert, eine Diskette (beziehungsweise Kassette) ins Laufwerk zu legen.

Sobald das alles erledigt ist, müssen Sie Ihre persönlichen Codezahlen eingeben, mit denen der Text »bearbeitet« werden soll. Für diese Codezahlen ist keine Begrenzung vorgesehen, da der Computer jedoch mit ihnen rechnen muß, empfiehlt es sich, ein manierliches Maß beizubehalten. (Mit Zahlen im dreistelligen Bereich sollte es keine Probleme geben.)

Jetzt teilt sich der Weg. Haben Sie vorhin im Minimenü eine 1 (für »codieren«) getippt, so können Sie jetzt Ihren Text eingeben. Dimensioniert ist ein Feld von 839 Zeichen, also gerade soviel, daß der Anweisungstext noch auf dem Bildschirm sichtbar bleibt und auch später, beim Decodieren, nichts vom Bildschirm nach oben »wegrutscht«. Wenn Sie übrigens Textteile mit »INST/DEL« löschen, so werden *alle* eingegebenen Zeichen, also sowohl die ursprünglichen Buchstaben als auch jedes »INST/DEL« mitgezählt und dann mitverschlüsselt. Damit lassen sich beim späteren Decodieren recht reizvolle Effekte erzielen. Probieren Sie's doch einfach mal aus!

Selbstverständlich können Sie die Texteingabe auch schon vor Erreichen des 839. Zeichens abschließen: Sie brauchen nur die Sterntaste (»*«) zu drücken. Wenn Sie den Stern in Ih-

```
10 FOR I=36864 TO 36935:READ A:S=S+A:POKE
   I,A:NEXT                                  <054>
20 IF S<>6828 THEN PRINT"FEHLER IN DATA'S!
   ":STOP                                    <034>
30 PRINT"{CLR,DOWN,5RIGHT}SYNCHRO-TEST FUE
   R DATASETTEN"                             <041>
40 PRINT"{3DOWN,5RIGHT}RUECKSPULEN UND 'PL
   AY' ODER"                                 <006>
50 PRINT"{6RIGHT}'PLAY'+'RECORD' DRUECKEN.   <119>
60 PRINT"{2DOWN}SIGNAL{SPACE,RVSON}A{RVOFF
   }USWERTEN ODER{SPACE,RVSON}S{RVOFF}CHRE
   IBEN? ";                                  <123>
70 POKE 204,0:GET A$:IF A$="A"THEN SYS 368
   96                                        <188>
80 IF A$<>"S"THEN 70                         <174>
90 SYS 36864                                 <126>
100 DATA 169,0,141,17,208,120,173,18,208,7
    4,74,74,74,41,8,9,7,133,1,141,24         <204>
110 DATA 212,74,74,74,141,32,208,76,6,144,
    234,120,169,7,133,1,169,0,141,17         <020>
120 DATA 208,141,32,208,173,13,220,41,16,2
    40,249,173,32,208,73,1,141,32,208        <046>
130 DATA 10,10,10,141,24,212,76,45,144,0,0
    ,1                                       <068>
```

Listing 1. »Synchro Justage« erlaubt Ihnen, Ihre Datasette auf fremde Kassetten einzustellen

```
5 PRINT"{CTRL-N,CLR,2DOWN,3SPACE}ELEKTRONI
  SCHES TAGEBUCH"                            <009>
10 POKE 808,254:PRINT"{2DOWN,3SPACE}MENUE:
   ":INPUT"{DOWN,3SPACE}1=CODIEREN, 2=DECO
   DIEREN";W                                 <045>
12 PRINT"{DOWN,3SPACE}BITTE DISKETTE EINLE
   GEN!{DOWN}"                               <105>
15 PRINT TAB(26)"TT.MM.JJJJ{2UP}":INPUT"{D
   OWN,18SPACE}DATUM ";D$                    <191>
18 PRINT"{DOWN,3SPACE}CODEZAHLEN X,Y,Z":IN
   PUT"{DOWN,3SPACE}DURCH KOMMA GETRENNT "
   ;A,B,C                                    <066>
20 DIM T(839),T$(839):X=0:IF W=2 THEN 70     <127>
25 PRINT"{CLR,DOWN,3SPACE}TEXTEINGABE (MAX
   . 839 ZEICHEN)":PRINT"{3SPACE}'*' = END
   E":PRINT"{DOWN}>";                        <126>
30 GET T$(X):IF T$(X)=""THEN 30              <202>
35 IF T$(X)="*"OR X=839 THEN 50              <096>
40 PRINT T$(X);:X=X+1:GOTO 30                <184>
50 OPEN 1,8,2,D$+",S,W":PRINT#1,X:FOR I=0
   TO X-1:B=B+(A*C):PRINT#1,ASC(T$(I))+C+B <218>
60 NEXT I:CLOSE 1:GOTO 99                    <143>
70 PRINT"{CLR,DOWN}>";:OPEN 1,8,2,D$+",S,R
   ":INPUT#1,X:FOR I=0 TO X-1:B=B+(A*C):IN
   PUT#1,T(I)                              . <019>
80 T(I)=(T(I)-B)-C:IF T(I)<0 OR T(I)>255 T
   HEN CLOSE 1:PRINT"{DOWN}FALSCHE EINGABE
   ":GOTO 99                                 <047>
90 PRINT CHR$(T(I));:NEXT I:CLOSE 1          <080>
99 CLR:POKE 808,237:END:REM -PTS,1985-       <004>

© 64'er
```

Listing 2. »Elektronisches Tagebuch«. Niemand hat Zutritt zu Ihren privaten Aufzeichnungen!

ren Texten verwenden wollen, setzen Sie an der entsprechenden Programmstelle (Zeile 35) einfach nur ein anderes ENDE-Zeichen ein.

Sobald die Eingabe beendet ist, schreibt der Computer zunächst die Anzahl der Zeichen in die Datei. Dann errechnet er Buchstabe für Buchstabe nach einem bestimmten, sich ständig systematisch verändernden Prinzip (Sie finden es in Zeile 50) eine bestimmte Zahl und speichert diese ebenfalls auf der Diskette (oder auf der Kassette).

Anschließend, und hier vereinigen sich die beiden Programmzweige wieder (Zeile 99), werden durch »CLR« alle Variablen, also der Text und Ihre Codezahlen, gelöscht. Nun wird die Blockierung von RUN/STOP und RESTORE aufgehoben und das Programm beendet.

Der zweite Zweig, das Decodieren, beginnt in Zeile 70. Hier wird genau umgekehrt verfahren wie im ersten Teil: Erst wird die Anzahl der zu decodierenden Zahlen gelesen, dann kommen die Zahlen selbst an die Reihe. Stück für Stück werden sie aus der Datei geholt, entschlüsselt, in Buchstaben umgewandelt und ausgegeben.

Befindet sich ein errechneter ASCII-Code jedoch nicht im vorgesehenen Bereich (0-255), sprich: waren die eingegebenen Codezahlen falsch, wird das Programm mit entsprechender Fehlermeldung abgebrochen (Zeile 80).

Es kann auch vorkommen, daß Sie einen unleserlichen Zeichensalat auf den Bildschirm bekommen. Auch dann waren die Codes falsch, jedoch hielt sich die Abweichung so sehr in Grenzen, daß es rechnerisch nicht einwandfrei überprüfbar war. Beachten Sie auch, daß Sie für jeden Tag nur einen Eintrag in Ihr »Tagebuch« vornehmen dürfen.

Tips zum Umschreiben

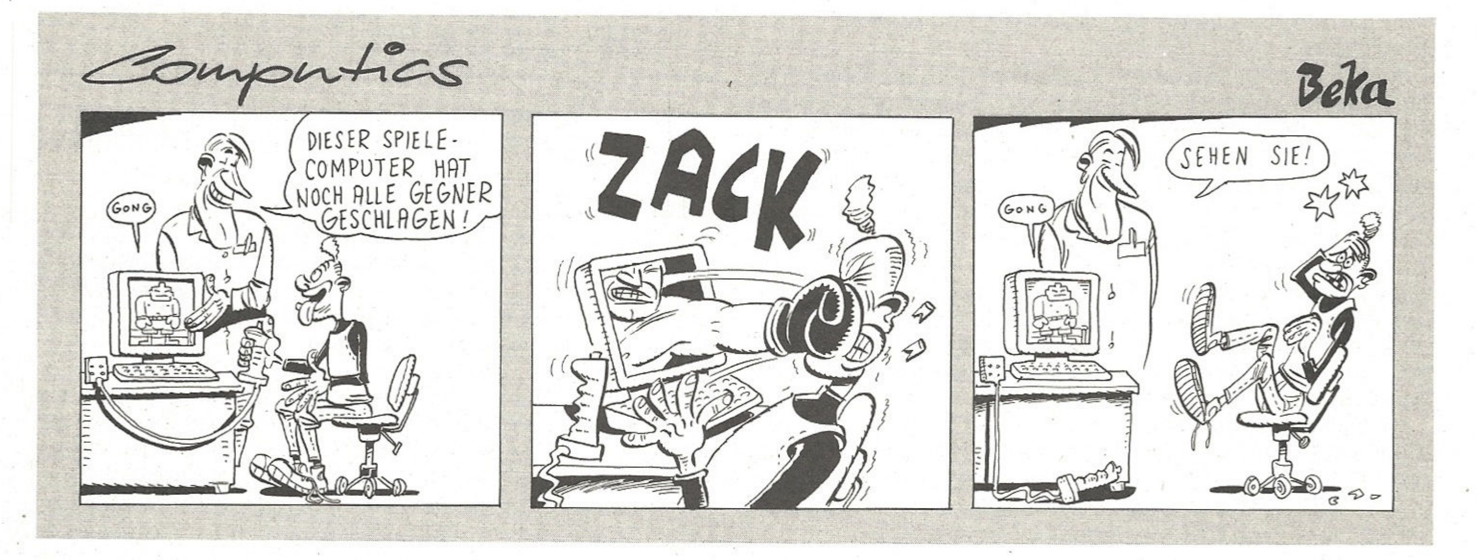
Kassettengebrauch: Der Umbau des Mikro-Tagebuchs auf Kassettenbetrieb dürfte sich recht einfach gestalten: Für die beiden OPEN-Befehle in den Zeilen 50 und 70 sähe die Änderung wie folgt aus:

Zeile 50: OPEN 1,1,1,D$

Zeile 70: OPEN 1,1,0,D$

Benutzer anderer Computer als dem C 64 müssen den POKE gegen RUN/STOP-RESTORE (in den Zeilen 10 und 99) gegebenenfalls ändern oder ganz weglassen.

(Peter T. Schmidt/tr)



Tips & Tricks für Profis

Hätten Sie gedacht, daß die rote LED an Ihrer Floppy ihre Helligkeit verändern kann? Oder daß ein kurzes Maschinenprogramm den Bildschirm auf den Kopf stellt? Auch diesmal haben wir für Sie einige Beiträge gesammelt, die selbst ausgefuchste Profis in Erstaunen versetzen werden.

Beginnen wollen wir diesmal mit einer wirklich außergewöhnlichen »Basic-Erweiterung«: Sie ermöglicht es, bis zu 252 Zeichen lange Basic-Zeilen einzugeben! Damit taucht unwillkürlich folgende Frage auf: Ist dies noch ein Einzeiler oder nicht?

Die Super-Einzeiler

Wer nicht will, daß sein Basic-Programm unnötig viel Speicher verbraucht, da er jedes freie Byte für Daten gebrauchen kann, oder wer will, daß sein Programm möglichst schnell ist, weil ihm jede Sekunde kostbar ist, für den ist es oft von Nutzen, mehrere Programmzeilen zu einer zusammenzufassen. Jede Zeile, die neu begonnen wird, verbraucht nämlich vier Byte mehr Speicherplatz, als wenn diese mit einem Doppelpunkt an die vorhergehende Zeile angehängt würde. Bei einem längeren Programm kann da schon einiges zusammenkommen. Außerdem kann der Interpreter Befehle in einer Zeile schneller abarbeiten, als wenn jeder Befehl in einer eigenen Zeile steht. Wenn diese sich dann noch in einer Schleife befinden, die vielleicht einige hundertmal durchlaufen wird, könnte man hier schon durch eine andere Anordnung Zeit sparen. Um das Programm allerdings nicht allzu unübersichtlich zu machen, sollte man nur solche Zeilen zusammenfassen, die auch logisch zusammengehören.

Wenn man dies nun beherzigen will, wird man allerdings durch den Basic-Editor ziemlich eingeschränkt, da dieser nur maximal 80 Zeichen in einer Zeile zuläßt. Der Interpreter würde aber auch längere Zeilen akzeptieren, nur kann man solche eben nicht eingeben.

Um dieses Manko zu beseitigen, habe ich EX-LINE geschrieben (siehe Listing 1). Mit diesem Programm ist es nun möglich, mit einer Länge von bis zu 252 Zeichen Basic-Zeilen einzugeben.

Nachdem man dieses Programm mit dem MSE eingegeben oder mit »LOAD "EX-LINE",8,1« von Diskette geladen hat, kann man es mit »SYS 49152« starten. Danach steht das Ausrufungszeichen als neuer Befehl zur Verfügung. Wenn man diesen Befehl eingibt, erscheint der Cursor in der linken oberen Ecke des Bildschirms. Nun kann man eine Programmzeile oder Befehle im Direktmodus eingeben. Dabei ist zu beachten, daß nach »RETURN« nur die ersten 6,5 Bildschirmzeilen übernommen werden, egal wo sich der Cursor dann befindet. Wenn man nach dem Ausrufungszeichen noch eine Zeilennummer angibt, wird vorher noch der Bildschirm gelöscht und die entsprechende Zeile gelistet, so daß man auch überlange Zeilen editieren kann.

Da sich das Programm nicht mit dem Checksummer verträgt, errechnet es noch eine Prüfsumme, die sich mit »PRINT PEEK(2)« auslesen läßt. Sie entsteht einfach durch Addition der ASC-Werte der eingegebenen Zeichen. In der Speicherzelle 2 steht dann das Low-Byte der Summe.

Wenn man die Befehle in abgekürzter Form eingibt, erhält man eine andere Prüfsumme, als wenn man sie ausschreibt. Die angegebenen Prüfsummen bei den folgenden Programmen beziehen sich immer auf die ausgeschriebene Form.

Programmbeschreibung

Das Programm ist komplett in Maschinensprache geschrieben und steht im Speicher ab $C000. Nach dem Start mit »SYS 49152« wird zuerst das Basic-ROM in den darunter liegenden RAM-Bereich kopiert und so verändert, daß der Eingabepuffer nicht mehr bei $0200, sondern bei $C200 liegt. Dies ist nötig, da dieser jetzt mehr Platz benötigt, als das Betriebssystem für ihn vorsieht.

Schließlich wird der Vektor für »Basic-Befehl holen« bei $0308 auf die neue Routine gelenkt. Diese prüft, ob das erste Zeichen ein Ausrufungszeichen ist. Wenn dies der Fall ist und noch eine Zeilennummer folgt, wird zunächst der Bildschirm gelöscht und die entsprechende Zeile gelistet. (Wenn keine Zeile angegeben war, wird dieser Teil übersprungen.)

Nachdem dann die Eingabe erfolgt ist, wird das RAM bei $A000 aktiviert, der Text im oberen Teil des Bildschirms in den Puffer bei $C200 kopiert, in Interpretercode umgewandelt und ausgeführt. Das ROM wird bei der nächsten Eingabe wieder eingeschaltet, so daß der Eingabepuffer wieder wie üblich bei $0200 liegt.

```
programm : ex-line          c000 c12e
----------------------------------------

c000 : a0 00 a2 a0 84 f7 86 f8   71
c008 : a2 20 b1 f7 91 f7 c8 d0   c3
c010 : f9 e6 f8 e6 fa ca d0 f2   c6
c018 : a9 c2 8d f5 a4 8d 84 a5   58
c020 : 8d ba a5 8d e7 a5 8d 06   13
c028 : a6 a9 c1 8d 13 a5 8d 16   86
c030 : a5 8d 24 a5 8d cd a5 8d   53
c038 : d0 a5 8d f1 a5 8d 0b a6   bd
c040 : 8d d2 aa 8d fd c1 a2 87   1a
c048 : a0 c0 8e 08 03 8c 09 03   ac
c050 : a9 d0 a2 7a 8d 28 a5 8e   27
c058 : 2b a5 8c 2c a5 a9 20 a2   6c
c060 : 82 8d 80 a4 8e 81 a4 8c   fe
c068 : 82 a4 a2 00 bd 1d c1 f0   93
c070 : 06 20 d2 ff e8 d0 f5 6c   01
c078 : 02 a0 b9 fc c1 91 5f 4c   97
c080 : 59 a6 a9 37 85 01 60 20   a0
c088 : 73 00 c9 21 f0 03 4c e7   ba
c090 : a7 a2 36 86 01 20 73 00   c6
c098 : b0 21 a2 ae a0 c0 8e 00   ai
c0a0 : 03 8c 01 03 48 a9 93 20   ea
c0a8 : d2 ff 68 4c a4 a6 20 ea   f3
c0b0 : e8 a2 8b a0 e3 8e 00 03   99
c0b8 : 8c 01 03 a9 13 20 d2 ff   38
c0c0 : 20 cf ff ad 00 04 48 a2   04
c0c8 : 00 86 02 a5 d4 48 a9 00   77
c0d0 : 85 d4 a9 13 20 d2 ff a9   78
c0d8 : 27 85 d0 bd 00 04 8d 00   04
c0e0 : 04 68 85 d4 20 cf ff 9d   d0
c0e8 : 00 c2 18 65 02 85 02 e8   22
c0f0 : d0 d9 8a 8d ff c2 a2 fc   9c
c0f8 : bd 00 c2 c9 20 d0 04 ca   cd
c100 : b8 50 f5 e8 a9 00 9d 00   8c
c108 : c2 68 8d 00 04 a9 0d a2   69
c110 : 03 20 d2 ff ca d0 fa 20   37
c118 : cf aa 4c 86 a4 0d 45 58   99
c120 : 2d 4c 49 4e 45 20 42 59   a1
c128 : 20 48 43 45 0d 00 ff ff   b7
```

Listing 1. Die Basic-Erweiterung »EX-LINE«

Um zu zeigen, was man mit diesem Programm alles in einer Zeile unterbringen kann, habe ich folgende drei »Einzeiler« geschrieben:

1. SOFT FLASH:

Dieser Einzeiler (Listing 2) bewirkt, daß die rote LED am Diskettenlaufwerk scheinbar stufenlos ein- und ausgeschaltet wird. Wenn man das Programm mit Hilfe von »EX-LINE« eingegeben hat, kann man mit »PRINT PEEK(2)« die Prüfsumme abfragen. Diese sollte 180 betragen.

Programmbeschreibung:

In der FOR-NEXT-Schleife wird ein Maschinenprogramm mittels »Memory-Write« in das RAM der Floppy ab $0500 geschrieben. Der Floppy-Befehl »UC« bewirkt dann, daß dieses gestartet wird. Das Maschinenprogramm schaltet die LED so schnell an und aus, daß dies für das Auge nicht sichtbar ist. Dabei ändert sich die Länge der Hell- und Dunkelphase, so

daß es scheint, als ob die Lampe langsam hell und dunkel würde.

Weil das Programm als Endlosschleife geschrieben ist, läßt sich das Laufwerk nicht mehr ansprechen. Wenn man die Floppy weiter verwenden will, muß man sie vorher aus- und anschalten.

```
1 OPEN 1,8,15:FOR I=0 TO 44:READ A:PRINT#1
  ,"M-W"CHR$(I)CHR$(5)CHR$(1)CHR$(A):NEXT:
  PRINT#1,"UC":DATA 169,254,170,32,21,5,20
  2,224,1,208,248,32,21,5,232,224,255,208,
  248,240,235,138,72,73,255,168,169,248,14
  1,0,28,202,208,248,169,240,141,,28,136,2
  08,248,104,170,96
```

© 64'er

Listing 2. »Soft Flash« läßt die rote LED der Floppy stufenlos an- und ausgehen. Zur Eingabe bitte Listing 1 verwenden.

2. STRICH-CURSOR:

Dieser Einzeiler (Listing 3) verwandelt den Cursor in einen Strich, der unter den Zeichen blinkt, so wie man es oft bei größeren Computern sieht. Man kann diese Routine verwenden, um seinen Programmen ein professionelles Aussehen zu verleihen (Prüfsumme: 186).

Programmbeschreibung:

Die ersten drei POKE-Befehle bewirken, daß der Bildschirm nach $CC00 verlegt wird und der Zeichensatz aus dem RAM ab $D000 gelesen wird. Dies ist notwendig, da kein Basic-Speicher verloren gehen soll und der VIC nur 16 KByte auf einmal adressieren kann.

Bevor man nun den Originalzeichensatz bei $D000 mit »POKE 1,3« lesbar machen kann, muß noch der Interrupt mit »POKE 56333,1« ausgeschaltet werden. Die erste FOR-NEXT-Schleife kopiert den Bildschirm an seine neue Position. Die zweite Schleife kopiert den Zeichensatz ins RAM. Dabei werden die reversen Zeichen so verändert, daß nur die unterste Reihe revers erscheint. Die folgenden POKEs bewirken schließlich, daß der Interrupt eingeschaltet wird und der I/O-Bereich bei $D000 wieder ansprechbar ist.

```
1 W=56333:Q=53248:Z=4↑5:POKE Q+24,52:POKE
  56576,0:POKE 648,204:POKE W,1:POKE 1,3:F
  OR I=0 TO 999:POKE 52224+I,PEEK(Z+I):NEX
  T:FOR I=0 TO Z:A=Q+I:B=A+2*Z:L=Z*((I AND
   7)=7):POKE A,PEEK(A):POKE B,PEEK(B):POK
  E A+Z,PEEK(A-L):POKE B+Z,PEEK(B-L):NEXT:
  POKE 1,7:POKE W,129
```

© 64'er

Listing 3. »Strich-Cursor«. Verleiht Ihrem Cursor ein äußerst professionelles Aussehen

3. UPSIDE-DOWN:

Dieses Programm habe ich in zwei Versionen geschrieben, da es als Einzeiler (Listing 4) sehr schwer abzutippen ist. Die zweite Version (Listing 5) wird mit dem MSE eingegeben (die Prüfsumme für den Einzeiler lautet 76). Beide Versionen können einfach mit »RUN« gestartet werden. Danach erscheint der ganze Bildschirm »auf den Kopf gestellt«. Dies bezieht sich auf *alle* Ein- und Ausgaben. Auch Zeichen, die mit »POKE« auf den Bildschirm gebracht werden, erscheinen an der entsprechenden anderen Stelle auf dem Kopf. Die meisten Programme vertragen sich damit gut, da nur der IRQ-Vektor verändert wird. Nach einem Reset oder RUN/STOP-RESTORE kann das Programm mit »SYS 300« wieder aktiviert werden.

Programmbeschreibung:

Das Maschinenprogramm, das bei dem Einzeiler mittels »PRINT« auf den Bildschirm gebracht wird, kopiert zunächst eine Interruptroutine in den Stack ab Adresse 300. (Dieser Bereich ist besonders geeignet, da er nach einem Reset nicht gelöscht wird und ein Programm dort nicht stört.) Dann wird der gesamte Zeichensatz auf den Kopf gestellt und in das RAM bei $D000 kopiert. Der VIC wird veranlaßt, den Zeichensatz aus $D000 und den Bildschirm aus $E000 zu lesen, und schließlich wird die Interruptroutine aktiviert, die 60mal in der Sekunde den Bildschirm umgekehrt nach $E000 kopiert. (H. C. Edelmann/tr)

```
1 PRINT"{HOME,RVSON,SPACE,RVOFF}*{RVSON}9{
  RVOFF}CD{RVSON}Y{RVOFF},A{RVSON}H{RVOFF}
  P{RVSON}Y{RVOFF}U{RVSON}){RVOFF}3{RVSON}
  E{RVOFF}A{RVSON}){RVOFF}@{RVSON}EY)PEU{S
  PACE,RVOFF}@{RVSON}1Y.9#{RVOFF}J&B{RVSON
  }JP@X{RVOFF}IG{RVSON}(%{RVOFF}B{RVSON}QY
  X{RVOFF}IG{RVSON}(HP++U%UI{SHIFT-SPACE}P
  ↑){RVOFF}7{RVSON}E{RVOFF}AL,AU{RVSON})DM
  {RVOFF}X{RVSON}P){RVOFF}@{RVSON}M{RVOFF}
  @{RVSON}-){RVOFF}C{RVSON}M{RVOFF}TC{RVSO
  N}){RVOFF}A{RVSON}M{RVOFF}UCX{SHIFT-SPAC
  E,RVSON}){RVOFF}@{RVSON}*(H={RVOFF}@D{RV
  SON}Y£I={RVOFF}@E{RVSON}Y£K={RVOFF}@F{RV
  SON}Y£{SHIFT-SPACE}={RVOFF}@G{RVSON}Y£*H
  £PB{RVOFF}L1{RVSON}N{RVOFF}":SYS 1024
```

© 64'er

Listing 4. »Upside-Down«. Dreht den Bildschirm um 180 Grad! Beachten Sie bitte die Eingabehinweise im Text.

```
programm : upside-down v2  0801 0896
--------------------------------------

0801 : 11 08 00 00 9e 32 30 37  c1
0809 : 32 20 ab 20 48 43 45 00  ee
0811 : 00 00 00 00 00 00 00 a0  53
0819 : 40 b9 5b 08 99 2c 01 88  1e
0821 : 10 f7 78 a9 33 85 01 a9  37
0829 : 00 85 f7 a9 d0 85 f8 a0  7d
0831 : 00 b1 f7 ae b9 a3 4a 26  0c
0839 : 02 ca d0 fa 98 49 07 a8  75
0841 : a5 02 91 f7 98 49 07 a8  8c
0849 : c8 d0 e6 e6 f8 a5 f8 c9  44
0851 : e0 d0 de a9 37 85 01 4c  c2
0859 : 2c 01 78 a9 84 8d 18 d0  10
0861 : a9 00 8d 00 dd a9 43 8d  c1
0869 : 14 03 a9 01 8d 15 03 58  c8
0871 : 60 a9 00 aa a8 88 bd 00  c1
0879 : 04 99 e8 e2 bd 00 05 99  03
0881 : e8 e1 bd 00 06 99 e8 e0  5c
0889 : bd 00 07 99 e8 df 88 e8  bd
0891 : d0 e4 4c 31 ea 20 8e b4  60
```

Listing 5. »Upside-Down V2«. Listing 4 als MSE-Dump

Print Shop und CP 80X

Probleme traten bei Print Shop und angeschlossenem Drucker (Melchers) CP 80X auf. Das lag daran, daß Print Shop nach jeder Druckzeile ein $0A sendet. Dies wird aber vom CP 80X laut Handbuch ignoriert. Abhilfe: Im Programmteil »Syslib« ($8800-$A0C0) die Adresse $9B31 von $0A auf $0D (= 13, entspricht Carriage Return + Linefeed) abändern und beim Setup den Druckertyp »Mannesmann Tally Spirit 80« anwählen. Seit dieser Änderung läuft Print Shop mit CP 80X einwandfrei. Die gesuchte Adresse befindet sich auf der (Original-) Disk im 20. Block, Adresse 5B von »Syslib«.

(Jürgen Römer/tr)

Disk-Optimizer

Ein nützliches Utility für Sparsame. Das Programm läuft auf allen Commodore-Computern, wenn man den DIM-Befehl in Zeile 60 dem verfügbaren Speicherplatz des jeweiligen Computers anpaßt.

Das Programm (Listing 6) ermöglicht die optimale Ausnutzung der Disketten. Man kann seine Programmsammlung auf

die kleinstmögliche Anzahl von Disketten verteilen und belastet damit den schmalen Geldbeutel vieler Computerfreaks nicht so stark. Ein Anwenderprogramm dieser Art ist bis jetzt nicht nur einzigartig, sondern nützt auch wirklich etwas — wird seinem Namen (Utility) also gerecht!

Anwendungsbeispiel:

Wir haben 13 Programme zur Verfügung, die auf möglichst wenige Disketten verteilt werden sollen. Sie haben folgende Blocklängen: 116,153,182,6,33,191,38,84,156,5,86,186,113. Nach dem Programmstart mit »RUN« geben wir die freien Blöcke einer Diskette ein (meistens 664) und anschließend die Blocklängen der 13 Programme in beliebiger Reihenfolge. Vertippt man sich, so kann man mit einer »0« oder einfach RETURN die jeweils letzte Eingabe berichtigen. Zum Abschluß der Eingabe muß die Zahl »999« folgen.

Jetzt sucht das Programm die Blöcke heraus, die addiert die vorher eingegebene Summe ergeben. Damit keine gleichen Zahlenfolgen auf dem Bildschirm erscheinen, sortiert das Programm solche Zahlenfolgen, die bereits schon einmal gefunden wurden, aus. Nach einer Weile sehen wir, daß es mehrere Kombinationen gibt, diese Programme platzsparend auf wenigen Disketten unterzubringen. Wir suchen uns die passenden aus und erhalten durch geschicktes Umkopieren der Programme auf eine oder mehrere Disketten brauchbaren Speicherplatz. Hätten Sie gedacht, daß man mit diesen 13 Programmen so viele verschiedene Möglichkeiten hat, eine Diskette randvoll zu füllen? Die Bedeutung der »Versuche«-Anzeige im Fenster oben links wird in der Programmbeschreibung erklärt. Das Programm sucht so lange nach neuen Folgen, bis Sie durch Drücken der START-/STOP-Taste »S« neu beginnen.

Da das Basic-Programm relativ langsam arbeitet, empfiehlt es sich, es zu compilieren. Erst bei 3- bis 4mal schnellerer Bearbeitungsgeschwindigkeit wird man es gern einsetzen. Die Programmservice-Diskette enthält neben der Basic- auch die compilierte Version.

(Martin Boels/tr)


```
40 AP=200                                        <068>
50 DL=20                                         <034>
60 DIM B(AP),E(AP),Z(AP),GS(AP,DL)               <222>
80 GOSUB 140                                     <088>
90 GOSUB 210                                     <058>
100 GOSUB 300                                    <060>
110 GOSUB 350                                    <150>
120 GOTO 90                                      <106>
140 PRINT"{CLR,DOWN,SPACE}DISK-OPTIMIZER:"
    :PRINT" EEEEEEEEEEEEEEE":INPUT"{DOWN,S
    PACE}SUMME =";S                              <023>
150 PRINT"{DOWN}BEI FEHLERN '0' EINGEBEN."
    :PRINT"ENDE: '999'":PRINT                    <149>
160 AZ=AZ+1:PRINT" ZAHL"AZ;:INPUT B(AZ)          <076>
170 IF B(AZ)=0 THEN AZ=AZ-2:PRINT"{2UP}";:
    GOTO 160                                     <067>
180 IF B(AZ)<>999 GOTO 160                       <230>
190 AZ=AZ-1                                      <132>
200 PRINT"{CLR}SUMME{3SPACE}:"S:PRINT"ZAHL
    EN{2SPACE}:";AZ:RETURN                       <154>
210 V=V+1:PRINT"{HOME,2DOWN}VERSUCH :"V          <177>
220 GET B$:IF B$="S"THEN RUN                     <099>
230 FOR I=1 TO T:B(Z(I))=E(I):NEXT:T=0:BS=
    0                                            <235>
240 Z%=RND(1)*AZ+1:IF B(Z%)=0 GOTO 240           <120>
250 BS=BS+B(Z%):T=T+1:E(T)=B(Z%):B(Z%)=0:Z
    (T)=Z%                                       <169>
260 IF BS>S GOTO 210                             <157>
270 IF BS=S THEN RETURN                          <016>
280 IF T=AZ THEN PRINT"{2DOWN}SUMME ALLER
    ZAHLEN IST KLEINER ALS";S:END                <197>
290 GOTO 240                                     <036>
300 G=T-1:FOR X=G TO 1 STEP-1                    <024>
310 F=0:FOR Y=1 TO G:IF E(Y)<=E(Y+1)GOTO 3
    30                                           <034>
320 F=Y:M=E(Y):E(Y)=E(Y+1):E(Y+1)=M              <088>
330 NEXT Y:G=F:IF F=0 THEN RETURN                <019>
340 NEXT X:RETURN                                <150>
350 FOR A=1 TO GS:FOR B=1 TO T                   <016>
```

Listing 6. »Disk-Optimizer«. Nützen Sie die vorhandenen Disketten-kapazitäten voll aus!

```
360 IF GS(A,B)<>E(B)THEN B=T:NEXT B,A:GOTO
     380                                         <011>
370 NEXT B:RETURN                                <004>
380 PRINT CHR$(19):GS=GS+1:FOR I=1 TO RZ+3
    :PRINT:NEXT                                  <140>
390 FOR I=1 TO T:GS(GS,I)=E(I):PRINT E(I);
    :NEXT                                        <243>
400 RZ=RZ+1:IF RZ<20 THEN RETURN                 <245>
410 GET A$:IF A$=""GOTO 410                      <219>
420 RZ=0:GOTO 200                                <150>
```


Listing 6. Schluß

Programmbeschreibung

Zeile	
40— 60:	Die vorgegebene Variablendimensionierung erfordert mindestens den Speicherplatz des C 64. Für kleinere Commodore-Computer muß knapper kalkuliert werden. AP gibt die größtmögliche Anzahl der zu verarbeitenden Programme an. DL gibt die größtmögliche Anzahl von Files auf einer Diskette an. Berechnung siehe im Listing.
80—120:	Hauptschleife mit Unterprogrammen
140—200:	Eingabe der Zahlen (Blocklängen) in B(AZ), AZ=Anzahl
210:	Anzeige der Versuche, die gewünschte Summe (664) zu erhalten.
220:	Abbruchmöglichkeit
230:	Initialisierung
240—250:	Zufälliges Aussuchen einer Blocklänge, doppelte ignorieren
260:	Diskette »überfüllt«, neu suchen
270:	Summe genau erreicht, Diskette voll
280:	Falls alle Blöcke addiert keine Diskette füllen: Abbruch
300—390:	Vergleich vorheriger Zahlenfolgen mit der jetzt gefundenen: falls neu: Bildschirmausgabe; sonst weitersuchen
400—420:	Scrollstop für den Bildschirm, falls unteres Ende erreicht; weiter mit Tastendruck

LoRes zu HiRes

Diese geniale Maschinenroutine (Listing 7) konvertiert den normalen Low-Resolution-Bildschirm, also den Bildschirm, der nach dem Einschalten aktiv ist, zu einem hochauflösenden Grafikbild und speichert dieses auf Diskette. Es kann sowohl die niedrig auflösende Grafik (Pixel-Grafik) als auch der normale Textbildschirm umgewandelt und gespeichert werden.

Bedienung

Nach dem Laden des Programms mit »`LOAD''LORES ZU HIRES'',8,1`« wird die Maschinenroutine mit »`SYS 828,''BILDNAME'',8`« aufgerufen. Die Parameter nach dem »`SYS 828,`« entsprechen genau den Angaben hinter dem normalen SAVE-Befehl. Da das Programm frei im Speicher verschiebbar ist, muß man nur die SYS-Startadresse auf den benutzten Speicherbereich anpassen. Ursprünglich liegt die Routine im Kassettenpuffer, wird also von keinem Basic-Programm gestört. Der Aufruf »`SYS 828,''NAME'',8`« kann sowohl im Programm- als auch im Direktmodus erfolgen. Im Direktmodus jedoch wird dann auch der eingegebene SYS-Befehl, der sich noch auf dem Bildschirm befindet, als Bestandteil des Textbildschirms gespeichert.

Anwendung:

Wird von einem LoRes-Grafikbild eine Hardcopy gemacht, so erscheint bei fast allen Druckern zwischen den Bildschirmzeilen ein weißer Streifen. Das ist beim Text ja auch sehr sinnvoll — wie sollte man ihn sonst auch lesen? Bilder werden durch diese Art des Druckes aber leider zerstört.

Nun hilft die »LoRes zu HiRes«-Routine: Man speichert sein Bild als hochauflösende Grafik, lädt es mit Hi-Eddi oder einem sonstigen Malprogramm wieder und kann nun mit die-

sem Programm eine »heile« Hardcopy erstellen. Wenn diese Hardcopy-Routine Bilder doppelt groß drucken kann (Hi-Eddi!), hat man erstmalig die Möglichkeit, einen LoRes-Bildschirm im Riesenformat auf Papier zu bringen.

Zweites Beispiel:

Wer schon einmal ein schönes Grafikbild mit Hi-Eddi erstellt hat, weiß, wieviel Arbeit das macht. Oft kann man die grobe, geometrische Struktur mit einem Maskengenerator (64'er, Ausgabe 8/85) erstellen. Da man sein Werk jetzt in Hi-Res übertragen kann, ist für die Feinarbeit mit Hi-Eddi schon gute Vorarbeit geleistet.

Drittens: Man kann sogar ein LoRes-Bild, das mit einem anderen Zeichensatz erstellt wurde, weiterverarbeiten! Der Konvertierungsroutine muß nur die Basisadresse des neuen Zeichensatzes mitgeteilt werden. Das High-Byte wird in die Speicherstelle 828+55 geschrieben, das Low-Byte muß $00 sein, braucht also nicht extra geändert zu werden.

Das Basic-Programm (Listing 8) ist äquivalent zu dem Maschinen. Interessant ist der Zeitvergleich: In Basic dauert die Hi-Res-Übertragung 2:13 min, in Maschinensprache 0,4 s! Die Basic-Version ist daher weniger zum Abtippen, sondern vielmehr zum leichten Verstehen des Algorithmus gedacht.

(Martin Boels/tr)

```
 programm : lores zu hires  033c 03ba
 ------------------------------------

033c : a9 00 8d 0e dc a9 33 85  fd
0344 : 01 a9 04 85 f8 a9 00 85  b3
034c : f7 85 f9 a9 20 85 fa a2  19
0354 : 04 a0 00 8a 48 98 48 a9  b7
035c : 00 85 fc b1 f7 85 fb 06  3c
0364 : fb 26 fc 06 fb 26 fc 06  63
036c : fb 26 fc 18 a5 fc 69 d0  46
0374 : 85 fc a2 00 a0 07 b1 fb  21
037c : 91 f9 88 10 f9 18 a5 f9  19
0384 : 69 08 85 f9 90 02 e6 fa  3c
038c : 68 a8 68 aa c8 d0 c4 e6  ac
0394 : f8 ca d0 bd a9 37 85 01  49
039c : a9 01 8d 0e dc 20 fd ae  0f
03a4 : 20 d4 e1 a9 20 85 fa a9  49
03ac : 00 85 f9 aa a9 f9 a0 40  b0
03b4 : 20 d8 ff 60 01 08 00 00  9d
```

Listing 7. »LoRes zu HiRes«. Verwandelt den Textbildschirm in eine HiRes-Grafik

```
10 POKE 56334,0:POKE 1,51                      <109>
20 FOR I=1024 TO 2023:BA=53248+PEEK(I)*8       <072>
30 FOR J=0 TO 7:POKE 8192+Z,PEEK(BA+J):Z=Z
   +1:NEXT J,I                                 <207>
40 POKE 1,55:POKE 56334,1                      <175>
```

Listing 8. Der grundlegende Algorithmus von Listing 7

Tips & Tricks zum C 128

Hauptsächlich neue POKEs und interessante Programmiertricks möchten wir Ihnen diesmal anbieten. Auch die Nutzung des kompletten deutschen Zeichensatzes im C 64-Modus ist nicht zu verachten.

Der C 128 ist im Kommen. Sein hervorragendes Basic macht ihn zu einem immer beliebteren Aufsteigercomputer. Aber gerade seine (im Vergleich zum C 64) komplexe, interne Struktur mit unter anderem drei Mikroprozessoren macht den C 128 zu einer Fundgrube für die Tips & Tricks-Ecke. Übrigens: Wenn Sie etwas Neues herausgefunden haben, schreiben Sie uns! Jeder brauchbare Trick wird veröffentlicht!

Programmieren in Z80-Assembler

Einige frischgebackene C 128-Besitzer möchten jetzt auch einmal in Z80-Assembler programmieren. Um den Z80-Mikroprozessor einzuschalten, muß das Bit 0 in Speicherstelle $fd505 auf 1 gesetzt werden. Doch wo beginnt der Prozessor seine Arbeit und was ist sonst noch zu beachten?

Zunächst müssen die System-IRQs gesperrt werden. Weiterhin sollte man sich in Bank 0 befinden (dort wird später das Z80-ROM von $0000 bis $1000 eingeblendet). Nachdem dies geschehen ist, kann man endlich das Bit 0 in $fd505 auf »High« legen. Nach dem Abfallen des Systemtaktes der 8502-CPU beginnt nun endlich der Z80 seine Arbeit ab der Adresse $0ffed. Hinter dem dort stehenden »NOP« findet sie einen Sprungbefehl nach $0008 in das nun eingeschaltene ROM. Das dort stehende Programm bootet normalerweise die CP/M-System-Diskette.

Um in eigene Programme zu gelangen, muß man vor dem Einschalten den Sprungbefehl verbiegen. Dazu schreibt man in die Speicherstelle $0ffee ein $c3 (entspricht dem »JP«-Befehl) und in die folgenden beiden Stellen die Startadresse im Low/High-Format.

Listing 1 zeigt ein Programm, das die oben genannten Punkte, außer dem Verändern des Sprungbefehls, für Sie ausführt. Nach dem Abarbeiten des Z80-Programms sollte man nach $0ffe0 springen. Dort wird die 8502-CPU reaktiviert (Näheres in Listing 2). Beide Programme stehen sofort nach Einschalten des Computers zur Verfügung.

Zum Schluß möchte ich noch auf einen Fehler im Betriebssystem hinweisen: Betätigt man, während der 40-Zeichen-Bildschirm gescrollt wird, die ASCII/DIN-Taste, so erscheint sehr oft in etwa der Mitte des Bildschirms ein Zeichensalat. Disassembliert ergeben die ASCII-Werte ein Programm, das auch in der erweiterten Zeropage zu finden ist.

(Frank Probst/tr)

»PRINT AT« oder »CHAR«?

Zunächst möchte ich zu dem Artikel »Entdeckungsreise durch den C 128« (64'er, Ausgabe 12/85) Stellung nehmen: Ein Simulieren des »PRINT AT«-Befehls, wie in diesem Artikel beschrieben, ist nicht nötig: Er ist im Basic 7.0 im Prinzip enthalten. Er wird mit dem »CHAR«-Befehl angesprochen, der nicht nur im Grafik-Modus wirksam ist. Format: char, Spalte, Zeile, Beispiel:

```
10 char,5,5,"hallo" : char,15,15,"....",1
15 char,20,20 : input "eingabe";a
```

Der eingebaute Sprite-Editor des C 128 besitzt außer den im Handbuch aufgeführten Anweisungen noch den Befehl »COPY«. Drückt man in der SPRDEF-Ebene die Taste »C«, so erscheint unten die Frage »COPY FROM ?«. Es wird dann die Nummer des Sprite angegeben (1-8), dessen Bitmuster in das momentan bearbeitete Raster kopiert werden soll. So lassen sich an verschiedenen Sprites geringfügige Änderungen

leicht anbringen. Hat man den Sprite-Editor aufgerufen, haben die Funktionstasten F1 bis F8 folgende Bedeutung: F1 = C, F2 = A, F5 = A, F6 = RETURN, F7 = RETURN, F8 = M

Der eingebaute Maschinensprache-Monitor kann zusätzlich zu den im Handbuch beschriebenen Funktionen die vier möglichen Zahlensysteme (Hexadezimal, Dezimal, Binär, Oktal) ineinander umrechnen und anzeigen. Dazu gibt man, wenn man den Monitor aufgerufen hat, nur die umzuwandelnde Zahl mit ihrem entsprechenden Kennsymbol ein (»$«, »+«, »&«, »%« siehe Handbuch, Anhang C-3).

Im Kapitel »Organisation der Zeropage« des Handbuches ist die Anfangsadresse des Tastaturpuffers nicht erwähnt. Dieser liegt bei 842 (dezimal) in der Bank 0 und hat normalerweise die Länge 10 (Länge in Speicherstelle dez. 2592, Bank 0). Der Tastaturpuffer ist in Verbindung mit der Speicherstelle, die die Anzahl der Zeichen im Tastaturpuffer enthält (dez. 208, Bank 0), wichtig, um vom Programm aus Funktionen auszuführen, die sonst nur im Direktmodus möglich sind (zum Beispiel die Übernahme von neuen Programmzeilen). Dieser Programmiertrick ist schon vom C 64 und VC20 her bekannt.

Neben den im Handbuch auf Seite 4 bis 6 erwähnten CTRL-Funktionen gibt es im C 128-Modus noch folgende:

```
CTRL - S   verbietet Bildschirmaufrollen (entspricht »NO
           SCROLL«-Taste)
CTRL - Q   erlaubt Bildschirmaufrollen, löst »Cursor down« aus
CTRL - T   entspricht DELETE-Taste
CTRL - M   entspricht RETURN-Taste
CTRL - J   entspricht LINE FEED-Taste
CTRL - R   Revers on
```

(Frank Mülheims/tr)

Neue SYS' und POKEs

Mit »SYS 65341« läßt sich beim C 128 ein Reset erreichen. Mit »SYS 65357« kommt man vom C 128-Modus in den C 64-Modus ohne Sicherheitsabfrage. Einen recht guten Programmschutz erreicht man mit folgenden POKEs:

```
POKE 802,0:POKE803, 224
```

In ein Programm eingebaut, lösen diese POKEs nach dem Drücken der RUN/STOP-Taste einen Reset aus. Wer Spaß an Bildschirmeffekten hat, sollte einmal folgendes Programm ausprobieren:

```
10 FOR I = 1 TO 15: SYS 51914: NEXT
20 FOR I = 1 TO 15: SYS 51900: NEXT
30 GOTO 10
```

(Andreas Ligendza/tr)

Programmschutz-POKEs

Wie man beim C 64 über »POKE 808,225« die RUN/STOP- und die RESTORE-Taste abschalten kann, geht dies beim C 128 über POKE 808,98. Der große Nachteil ist, daß damit die interne Uhr (TIME,TI,TIME$,TI) unbrauchbar wird. Doch durch einen Reset und gleichzeitiges Drücken von RUN/STOP kann man in den Monitor gelangen, diesen verlassen, und ans Programm kommen, da es nicht gelöscht ist. Dies umgeht man über BANK 1:POKE 65528,3. Nach diesem POKE führt ein Reset nur zum Absturz. Dieser POKE funktioniert selbstverständlich auch ohne das Abschalten von RUN/STOP-RESTORE, aber beide POKEs zusammen ergeben einen recht guten Schutz. Wer auf TIME$ (TI$) und TIME (TI) nicht verzichten will, kann mit POKE792,51:POKE793,255 die Tastenkombination RUN/STOP-RESTORE abschalten, die RUN/STOP-Taste alleine funktioniert aber weiterhin. Die letzten POKEs eignen sich vor allem dann, wenn man ein Basic-Programm compiliert: nach dem Compilieren sind sie ohnehin gegen RUN/STOP (ohne RESTORE) geschützt, und durch die beiden letzten POKEs wird auch noch die Kombination RUN/STOP-RESTORE und/oder durch BANK1:POKE65528,3 der RESET verhindert.

(Florian Müller/tr)

Einfache Spritesteuerung

```
10 SPRDEF:SPRITE 1,1
20 J=(JOY(2)-1)*45:IFJ=-45THENMOVSPR1,0#0:GOTO20:
   ELSE:MOVSPR1,J#7:GOTO20
```

Dieser Einzeiler ermöglicht eine einfache (aber schnelle) Joystickabfrage. Statt »IF...THEN«-Anweisungen werden die Joystickrichtungen in Grad umgewandelt und später von einem »MOVSPR x,x#x«-Befehl verarbeitet.

Zum Programm:

In Zeile 10 wird das Sprite definiert (SPRDEF) und eingeschaltet.

In Zeile 20 wird in der Variablen J die Richtung des Joysticks —1*45 (360 Grad / 8 Richtungen = 45 Grad) abgelegt. Ist J=—45 (bei Nullstellung), wird das Sprite gestoppt und das Programm springt wieder nach Zeile 20.

Ist J nicht —45, bewegt sich das Sprite in Richtung J (mit »MOVSPR 1,j#Geschwindigkeit« (die Geschwindigkeit ist beliebig von 2-15 wählbar). Zum Schluß startet das Programm erneut von Zeile 20.

(Thorsten Wanschura/tr)

Deutscher Zeichensatz im C 64-Modus

Wenn man im C 64-Modus die »CAPS-LOCK«-Taste drückt, stehen alle internationalen Zeichen zur Verfügung, die sonst nur vom C 128-Modus aus erreichbar sind. Dabei werden einige Grafikzeichen mit den internationalen Zeichen belegt. Die CHR$ Tabelle sieht nun folgendermaßen aus:

CHR$ (172) = é	CHR$ (187) = ä	CHR$ (174) = è	CHR$ (123) = Ä
CHR$ (177) = µ	CHR$ (188) = ö	CHR$ (178) = à	CHR$ (124) = Ö
CHR$ (179) = ù	CHR$ (189) = ü	CHR$ (180) = â	CHR$ (125) = Ü
CHR$ (181) = ê	CHR$ (190) = ß	CHR$ (182) = î	CHR$ (191) = ^
CHR$ (183) = ô	CHR$ (192) = à	CHR$ (184) = û	CHR$ (64) = §
CHR$ (185) = √	CHR$ (96) = á	CHR$ (186) = ∑	

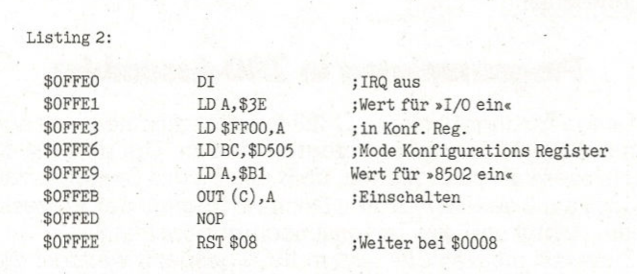
(Carsten Schmidt/tr)

```
Listing 1:

$0FFD0     SEI          ;IRQ aus
$0FFD1     LDA #$3E     ;Wert für »I/O ein«
$0FFD3     STA $FF00    ;in Konf. Register
$0FFD6     LDA #$B0     ;Wert für »Z80 ein«
$0FFD8     STA $D505    ;in Mode Konf. Reg. Kurz: Einschalten
$0FFDB     NOP
$0FFDC     JMP $1100    ;muß auf eigene Routine verbogen werden

Wichtig: Die Z80-CPU beginnt ihre Arbeit bei $0FFED!
```

Listing 1. So wird die Z80-CPU eingeschaltet



```
Listing 2:

$0FFE0     DI            ;IRQ aus
$0FFE1     LD A,$3E      ;Wert für »I/O ein«
$0FFE3     LD $FF00,A    ;in Konf. Reg.
$0FFE6     LD BC,$D505   ;Mode Konfigurations Register
$0FFE9     LD A,$B1      Wert für »8502 ein«
$0FFEB     OUT (C),A     ;Einschalten
$0FFED     NOP
$0FFEE     RST $08       ;Weiter bei $0008

Wichtig: Die 8502-CPU beginnt ihre Arbeit bei $0FFDB!
```

Listing 2. Ein Z80-Programm, um die 8502-CPU wieder einzuschalten

Tips & Tricks zum C 16

Auch zu Commodores Kleinstem haben wir einige interessante Programmierhinweise für Sie herausgesucht. Neben einigen nützlichen POKEs finden Sie ein Programm, um den Zeichensatz nach Ihren Wünschen zu ändern.

Der Zeichensatz besteht beim C 16 aus nur 128 Zeichen. Die anderen 128 Zeichen sind die inverse Darstellung der ersten 128 Zeichen. Ein Zeichensatz kann also alle 1024 Bytes stehen (zum Beispiel ab den Adressen 12288, 13312, 14336, 15360).

Nun die Befehle, um ihn zu verändern:

1. »POKE 65298, PEEK (65298) AND NOT 4«
2. »POKE 65299, Anfang Zeichensatz/256«

Nach einer Fehlermeldung sind allerdings keine Zeichen mehr zu erkennen. Sie erscheinen erst dann wieder, wenn wieder ein Befehl eingegeben wird.

Listing 1 zeigt ein Programm zum Kopieren des Zeichensatzes vom ROM ins RAM. Startadresse des Zeichensatzes ist 15360.

Farben auf den Farbtasten verändern

Die Farben der Farbtasten (CTRL- oder Commodore-Taste zusammen mit einer Zahlentaste) sind in den Adressen 275 bis 290 gespeichert. Sie können beliebig geändert werden. Die entsprechenden Steuerzeichen bleiben allerdings gleich!

Beispiel: POKE 275,69.
Statt Schwarz liegt auf der ersten Farbtaste jetzt Grün.

Wiederholfunktion der Tasten

POKE 1344,0:	Wiederholfunktion aus, nur »CRSR«, »SPACE«, »INST/DEL« haben Wiederholfunktion
POKE 1344,64	keine Taste hat Wiederholfunktion
POKE 1344,128	alle Tasten haben Wiederholfunktion (Normalzustand).

Tastaturpuffer

Die Adressen 1319 bis 1328 enthalten die Codes der Tasten, die nicht unmittelbar ausgeführt werden können.

239 enthält den Zähler für den Tastaturpuffer, der angibt, wieviel Codes im Tastaturpuffer abgelegt sind.

Hier ein Beispiel, bei dem ein Programm nachgeladen wird (in einem Programm)

```
POKE 1319,13 : POKE 1320,13 : POKE 239,2
PRINT "DLOAD" : CHR$ (34)
PRINT "RUN"
```

Tastaturabfrage

In der Adresse 198 wird der Code der gerade gedrückten Taste abgelegt. Wenn keine Taste gedrückt wird, ist PEEK (198) gleich 64. Mit PEEK (198) können auch die Funktionstasten abgefragt werden.

Abfragen der »SHIFT«-, »Commodore«- und »CTRL«-Tasten

Mit PEEK (1347) können diese Tasten abgefragt werden.

(Ulrich Käfferbitz/tr)

```
10 FORT=0TO56
20 READA
30 POKE15000+T,A
40 NEXT
80 SYS1500
62900 DATA 169,0,141,248,7,133,208,133
62910 DATA 210,169,208,133,209,169,60,133
62920 DATA 211,141,19,255,160,0,177,208
62930 DATA 145,210,200,208,249,230,209,230
62940 DATA 211,165,211,201,64,208,237,169
62950 DATA 192,141,18,255,169,59,133
62960 DATA 52,133,56
62970 DATA 169,246,133,51,133,55,96
```

Listing 1. Verschiebt den Zeichensatz ins RAM.

Nützliche Speicherstellen

— WAIT 1,192 wartet darauf, daß eine Taste am Recorder gedrückt wird.
— Der Funktionstastenspeicher steht in den Adressen $0567 bis $05E6
— Die Einsprungadresse für die USR-Funktion steht in #1281/1282 = $0501/0502

— Basic-Start	43/44 (Low-/Highbyte)
— Anfang der Variablen	45/46 (Low-/Highbyte)
— Anfang der Felder	47/48 (Low-/Highbyte)
— Ende der Felder	49/50 (Low-/Highbyte)
— Stringspeicher Ende	51/52 (Low-/Highbyte)
— Anfang der Strings	53/54 (Low-/Highbyte)
— Basicspeicher Ende	55/56 (Low-/Highbyte)

— Der Ton-Chip des C 16 läßt sich in Assembler wie folgt programmieren: Soundregister ist Adresse 65297. Jedem Bit kommt eine bestimmte Steuerfunktion zu:

Bit 0 bis 2	= #0 bis #7 Lautstärke = VOL 0 bis VOL 6
Bit 3	= #8 volle Lautstärke (wenn Bit 3 gesetzt ist, sind Bit 0-2 ohne Bedeutung)
Bit 4	= #16 Stimme 1 an (=1) / aus (=0)
Bit 5	= #32 Stimme 2 an (=1) / aus (=0)
Bit 6	= #64 Stimme 2 Rauschen (=1)
Bit 7	= #128 Bit gesetzt = kein Ton

Der Ton ist in 10 Bits codiert: mögliche Werte: #0 bis #1023

Frequenzen		Low-Byte	High-Byte (0,1)
	Stimme 1	65294	65298
	Stimme 2	65295	65296

— Register 65286:	Durch Löschen der Bits 0/1 läßt sich der Bildschirm um bis zu drei Rasterzeilen nach oben verschieben.	
	Bit 2 gesetzt	Bildschirm vier Rasterzeilen nach unten
	Bit 3 gelöscht	Bildschirm vier Rasterzeilen nach unten
	Bit 4 gelöscht	Bildschirm ausgeblendet
	Bit 5 gesetzt	Char-Speicher verschoben
	Bit 6 gesetzt	kein Cursor sichtbar
— Register 65287	Bit 0/1 setzen	Verschieben des Bildschirms um bis zu drei Rasterzeilen nach rechts.
	Bit 4 gesetzt	Mehrfarbmodus ein
	Bit 5 gesetzt	Videochip stoppt (keine Bilderzeugung mehr)
	Bit 6 gesetzt	Programmierbare Bildstörung
	Bit 7 gesetzt	38 Zeichen/Zeile
— POKE 65290, 162	sperrt Tastatur	
— POKE 65290, 163	entriegelt Tastatur	
— Register 65301	Hintergrundfarbe	Bit 0—3 (Low-Nibble) Farbe
— Register 65305	Rahmenfarbe	Bit 4—6 (High-Nibble) Intensität

(Frank Plachetta/tr)



»Apfelmännchens« Diashow

Mit wenig Aufwand kann man die Grafiken des »Apfelmännchens« zu einer effektvollen und abwechslungsreichen Diashow zusammenfassen.

Natürlich darf bei einem Programm wie dem »Apfelmännchen« (64'er 11/85, Seite 80) der Vorführeffekt nicht zu kurz kommen. Um diesen zu erreichen, verändern Sie die Zeilen des Originalprogramms wie in Listing 1 angegeben und ergänzen das Programm mit Listing 2. Die Bilder müssen dazu die unten aufgeführten Namen haben. Bitte beachten Sie das Leerzeichen vor jeder Zahl:

Bild 1.PIC
Bild 2.PIC
Bild 3.PIC
...
Bild 9.PIC
Bild 10.PIC

Die Anzahl der Bilder wird in Zeile 11000 festgelegt. Um zum Beispiel die Anzahl auf 20 festzulegen, müßte dort stehen:

```
11000 N=N+1:IFN=21THEN RETURN
```

Das nächste Bild wird jeweils durch Druck auf die Space-Taste gezeigt.

(O. Hobert/og)

```
317 PRINT"{DOWN}"TAB(10)"(8){2SPACE}DIA-SH
    OW"                                   <229>
340 GET A$:IF A$<"1"OR A$>"8"THEN 340     <018>
360 ON VAL(A$)GOSUB 1000,2000,3000,4000,50
    00,5500,7000,10000                    <184>
```

© 64'er

Listing 1. Die Veränderungen ...

```
10000 REM DIA-SHOW                              <177>
10010 PRINT"{CLR,DOWN}*** DIA-SHOW ***"         <108>
10020 PRINT"{DOWN}FARBEN{9SPACE}: FUNKTION
      STASTEN"                                  <228>
10030 PRINT"{DOWN}MENUE{10SPACE}: '←'"          <150>
10040 PRINT"{DOWN}NAECHSTES BILD : 'SPACE'
      "                                         <229>
10050 PRINT"{2DOWN}WEITER MIT TASTE"            <105>
10060 POKE 198,0:WAIT 198,1:GET A$              <078>
10070 N=0                                       <117>
10080 REM LADEN                                 <098>
11000 N=N+1:IF N=10 THEN RETURN                 <208>
11005 N$=STR$(N)                                <200>
11010 NB$="BILD"+N$+".PIC"                      <057>
11020 FOR I=0 TO LEN(NB$)-1                     <046>
11030 POKE BN+I,ASC(MID$(NB$,I+1,1))            <143>
11040 NEXT                                      <048>
11050 POKE BN+16,LEN(NB$)                       <071>
11060 SYS LO                                    <090>
11070 REM EFFEKT                                <114>
11090 SYS M1                                    <159>
11100 IF A$="{F5}"THEN POKE C2,(PEEK(C2)+1
      )AND 15                                   <020>
11110 CP=PEEK(C3):POKE C3,PEEK(C2):POKE C2
      ,PEEK(C1):POKE C1,CP                      <085>
11120 GET A$:IF A$="←"THEN 11190                <201>
11125 IF A$=" "THEN GOTO 11200                  <112>
11130 IF A$="{F1}"THEN POKE C0,(PEEK(C0)+1
      )AND 15                                   <009>
11140 IF A$="{F3}"THEN POKE C1,(PEEK(C1)+1
      )AND 15                                   <168>
11150 IF A$="{F5}"THEN POKE C3,(PEEK(C3)+1
      )AND 15                                   <090>
11160 IF A$="{F7}"THEN POKE C3,(PEEK(C3)+1
      )AND 15                                   <229>
11170 SYS SC                                    <064>
11180 GOTO 11110                                <078>
11190 SYS M0:RETURN                             <221>
11200 SYS M0:GOTO 11000                         <016>
```

© 64'er

Listing 2. ... und die Ergänzungen für eine Diashow mit den Grafiken des »Apfelmännchens«

Autochange für C 128

Mit diesem Utility erkennt Ihr C 128 automatisch beim Boot-Versuch, in welchen Modus — C 64 oder C/PM — er springen muß.

Der C 128 sucht nach dem Einschalten zuerst auf der Diskette, ob er einen Boot-Sektor vorfindet. Ein Boot-Sektor ist ein Datenblock auf der Diskette, der beim Einschalten des Computers automatisch geladen und ausgeführt wird. Wird zum Beispiel ein CP/M-Boot-Sektor gefunden, so versucht er anschließend, das CP/M-Betriebssystem hochzufahren. Man kann diesen Sektor dahingehend umändern, daß er bei C 64-Disketten automatisch die Kontrolle an das C 64-Betriebssystem abgibt. Das Programm (Listing 1) generiert einen Boot-Sektor auf einer beliebigen, im Commodore-Format beschriebenen Diskette. Hierdurch wird der C 128 befähigt, eine C 64-Diskette zu erkennen und automatisch nach dem Einschalten in dessen Modus zu springen. Beim Erstellen dieses Startups (Befehlssektors) werden keine eventuell vorhandenen Dateien auf der Diskette geschädigt. Außerdem erkennt das Programm einen möglicherweise im C 128-Modus installierten Boot-Sektor und weist, wenn bestehend, gesondert darauf hin.

Zum Programm:

In den Zeilen 190 bis 230 wird auf das eventuelle Vorhandensein eines Boot-Sektors auf Spur 1/Sektor 0 geprüft. Sollte die Abfrage negativ ausfallen, wird die Installierung fortgesetzt, indem der Inhalt des späteren Startsektors im Blockpuffer aufgebaut wird (Zeile 380 bis 400). Der Blockpuffer enthält fortlaufend die Kennung »CBM«, gefolgt von vier Null-Byte und dem Text, der beim Booten ausgegeben wird (in NAME$ enthalten). Es schließen sich außerdem noch zwei Null-Byte und der eigentliche Sprungbefehl an, der direkt in den C 64-Modus umschaltet. Zeile 410 schreibt den Autochanger endgültig auf die Diskette und kennzeichnet ihn, wenn nötig, in der BAM als belegt.

(Manfred Bauer/dm)

```
10 LC=0:COLOR0,1:COLOR4,1:COLOR5,2
100 NAME$="{CLR,7RIGHT,7DOWN}C64 DISKETT
E"
110 PRINT"{CLR,5DOWN,9RIGHT}AUTOMATISCHE
R WECHSEL"
120 PRINT"{DOWN,11RIGHT}IN DEN 64'ER MOD
US"
130 OPEN15,8,15,"I0":A$=CHR$(0)
140 IFDS<>0THENBEGIN
150 PRINT"{2DOWN,RIGHT}DISKETTEN FEHLER:
 ";DS$
160 CLOSE8:CLOSE15
170 PRINT"{2DOWN,RIGHT}ABBRUCH":END
180 BEND
190 OPEN8,8,8,"#"
200 PRINT#15,"U1:8 0 18 0":PRINT#15,"B-P
";8;5
210 GET#8,BA$
220 PRINT#15,"U1:8 0 1 0"
230 FORI=0TO24:GET#8,S$:BL$=BL$+CHR$(ASC
(S$)):NEXT
240 IF(ASC(BA$)AND1)=0THENBEGIN
250 IFLEFT$(BL$,3)="CBM"THENPRINT"{2DOWN
,6RIGHT}BOOT-SECTOR SCHON VORHANDEN"
270 IFLEFT$(BL$,3)<>"CBM"THENPRINT"{2DOW
N,5RIGHT}PROGRAMM LIEGT AUF BOOT-SECTOR"
280 PRINT"{2DOWN,4RIGHT}INSTALLIERUNG FO
RTSETZEN? (J/N)" :LC=1
290 GETKEYX$
300 IFX$<>"J"THENBEGIN
310 CLOSE8:CLOSE15
320 PRINT"{2DOWN,5RIGHT}ABBRUCH":END:BEN
D
330 BEND
340 PRINT"{DOWN,5SPACE}INSTALLIEREN DES
BOOT-SEKTORS{CTRL-@}H{GREEN}Y{CTRL-A,LIG
.GREEN}"    BITTE ENTFERNEN SIE EINEN EV
TL.
350 PRINT"{5SPACE}VORHANDENEN SCHREIBSCH
UTZ UND{CTRL-@}&{BLUE}H{CTRL-A,LIG.GREEN
}"         DRUECKEN SIE EINE TASTE
370 GETKEYX$
380 PRINT#15,"B-P 8 0"
390 PRINT#8,CHR$(67);CHR$(66);CHR$(77);A
$;A$;A$;A$;NAME$;A$;
400 PRINT#8,A$;CHR$(DEC("20"));CHR$(DEC(
"4D"));CHR$(DEC("FF"));A$
410 PRINT#15,"U2:";8;0;1;0
415 IFLC=0THENPRINT#15,"B-A 0 1 0"
420 CLOSE8:CLOSE15
430 PRINT"{4SPACE}BOOT-SECTOR INSTALLIER
T":END
```

©64'er

Listing 1. Autochange 128/64. Bitte im C 128-Modus eingeben.

Das Maß der Dinge

Das exakte Ausmessen eines Unterprogrammes ist eine große Hilfe, um Laufzeiten zu verringern. Meist steht jedoch keine hinreichend genaue Uhr zur Verfügung. Wir bieten Ihnen zwei Lösungen an, eine für Maschinensprache- und eine für Basic-Programme.

Um die Laufzeiten von Programmen genau messen zu können, reicht die Stoppuhr für den 100-Meter-Lauf nicht mehr aus. Vor allem bei Maschinenprogrammen, bei denen die menschliche Reaktionsfähigkeit weit hinter der geforderten Genauigkeit bleibt. Doch auch die eingebaute interruptgesteuerte Uhr TI$ im C 64 versagt, wenn es um das Auszählen sehr kurzer Maschinenprogramme geht. Ladevorgänge können damit überhaupt nicht erfaßt werden, da TI$ während des Ladens stillsteht. Abhilfe schaffen hier die Timer der CIAs.

Taktzyklen zählen

Um Maschinensprache-Programme zu messen, ist es am besten, die Taktzyklen zu zählen. Dies liefert einen gut zu vergleichenden Wert. Geben Sie zunächst das Programm »Taktzyklen« (Listing 1) mit dem MSE ein und speichern Sie es. Aufgerufen wird die Routine durch den Befehl

SYS 49152,Startadresse <RETURN>

Die anzugebende Startadresse entspricht der des zu untersuchenden Maschinenprogrammes. Dieses Unterprogramm wird nun ausgeführt und die dafür benötigte Zeit bestimmt. Während dieses Ablaufes ist der Bildschirm ausgeblendet, er nimmt die Farbe des Rahmens an. Am Ende des zu messenden Programmes muß ein RTS stehen. Nach Programmende gibt das Programm den gemessenen Wert aus und springt in den Direkt-(READY.-)Modus. Der angezeigte Zahlenwert drückt die Anzahl der Systemtakte aus, die während der Abarbeitung des Maschinen-Unterprogrammes durch den Mikroprozessor verstrichen sind. Da ein solcher Takt etwa 1,015 Mikrosekunden dauert, läßt sich durch Multiplizieren mit diesem Faktor die benötigte Zeit in Mikrosekunden errechnen. Bei eingeschaltetem Bildschirm vergrößert sich das Ergebnis um etwa 5 bis 6 Prozent.

Beschreibung des Programms »Taktzyklen«

Die beim Aufruf des Programms übergebene Startadresse des zu messenden Unterprogramms wird als Operand für einen Sprungbefehl gespeichert. Anschließend wird die Möglichkeit der Unterbrechung durch einen IRQ ausgeschaltet, weil sonst ein falsches Resultat geliefert werden kann. Nach dem Ausblenden des Bildschirms lädt das Programm beide Timer des CIA 2 mit dem Maximalwert und koppelt sie miteinander. Sobald die Zähler gestartet sind, erfolgt der Sprung in das Unterprogramm, nach dessen Ende die Zähler sofort wieder gestoppt werden. Schließlich werden neben der Einrichtung des Normalzustandes — den Bildschirm und den IRQ betreffend — die Timer-Bytes ausgewertet. Dazu invertiert das Programm sämtliche Bits der von den Timern gelieferten Werte, da beide Registerpaare heruntergezählt werden. Zusätzlich werden noch elf Takte subtrahiert, die zum Starten beziehungsweise Stoppen der Timer und zum Aufruf des Unterprogramms nötig sind. Damit erhält man einen Meßbereich, der zwischen 1 und 2^{32}-11 Taktzyklen (entsprechend einer Zeit von 1 μs bis 1 Stunde 12 Minuten) liegt.

Laufzeit-Meß-System »LMS«

Das LMS wurde entwickelt, um die Laufzeit von Basic-Programmen zeilenbezogen messen zu können. Weiterhin sollte

die Anzahl einzelner Aufrufe von Programmzeilen festgestellt werden können.

Einsatzmöglichkeiten

Das LMS ist im wesentlichen als Werkzeug für folgende Anwendungsfälle konzipiert.

1. Performance-Verbesserung von Basic-Programmen

Durch den Einsatz des LMS ist sofort erkennbar, wo die zeitmäßig kritischen Stellen eines Programms sind. Oft kann durch einige Veränderungen von Basic-Statements wertvolle Laufzeit eingespart werden. Sollte das nicht ausreichen, so ist der Einsatz von Maschinenroutinen angebracht.

Die Erfahrung zeigt, daß es selten notwendig ist, ein Programm vollständig in Assembler zu schreiben. Meist bringt ein begrenzter Einsatz von Maschinenroutinen an zeitkritischen Stellen ausreichende Geschwindigkeit unter Beibehaltung der Vorteile von Basic. Diese sind vor allem wesentlich bessere Lesbarkeit und Wartung sowie geringerer Erstellungsaufwand.

2. Testhilfe

Das Laufzeit-Meß-System stellt die Anzahl der Aufrufe von Basic-Zeilen fest. Es ist daher damit leicht überprüfbar, ob Programmzeilen überhaupt durchlaufen werden und wenn ja, ob die Anzahl der Aufrufe stimmt.

Mit dem LMS können auch Statements gemessen werden, die mit TI$ — abgesehen von der geringen Genauigkeit von TI$ — zu erfassen sind (zum Beispiel INPUT#).

Bedienungsanleitung

Geben Sie zuerst die Listings 3 bis 5 ein und speichern Sie diese auf einer Diskette. Nach dem Laden und Starten des Programms »LMS« erscheint nach kurzer Zeit ein Menü mit folgenden Wahlmöglichkeiten:

Messen (m)

Durch die Auswahl der Funktion »Messen« wird das eigentliche Meßprogramm aktiviert. Dies ist immer erkennbar an der roten Rahmenfarbe. Es wird eine Einschaltmeldung geschrieben und danach in den READY-Modus verzweigt. Jetzt muß das zu messende Programm eingegeben oder geladen werden.

Bei jedem Start des Programms werden für jede Programmzeile Laufzeit und Aufrufanzahl gespeichert. Wichtig ist, daß das Programm mit RUN gestartet wird. Ein Start mit GOTO oder CONT ist möglich, führt aber nicht zur Messung.

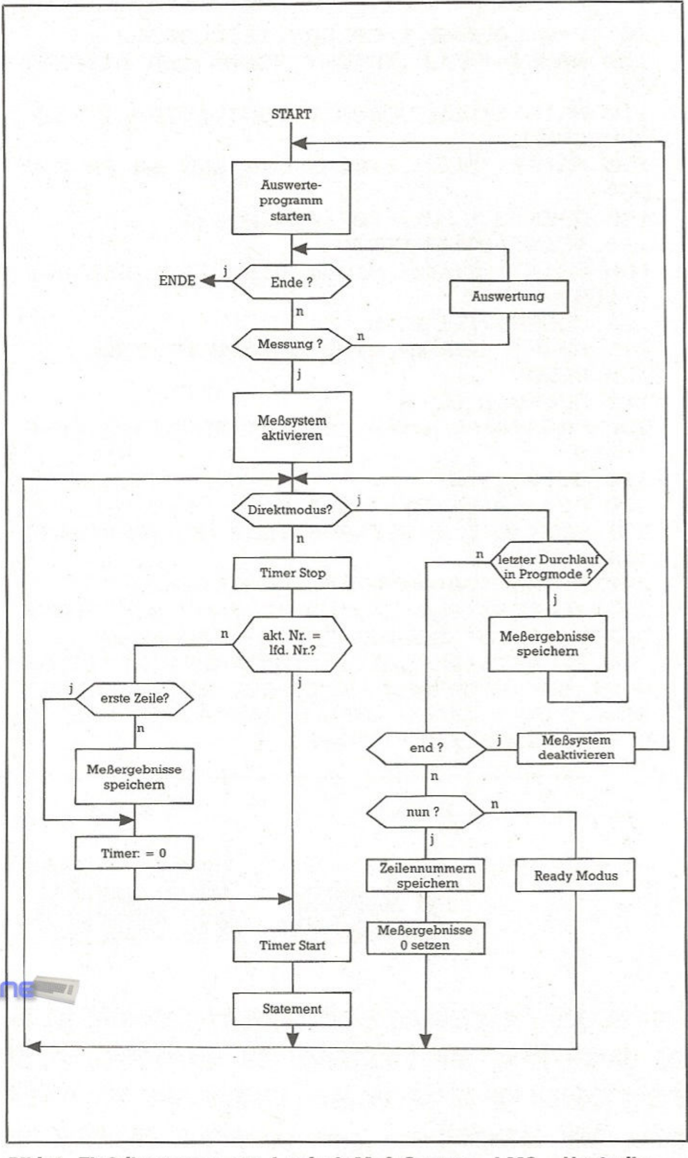
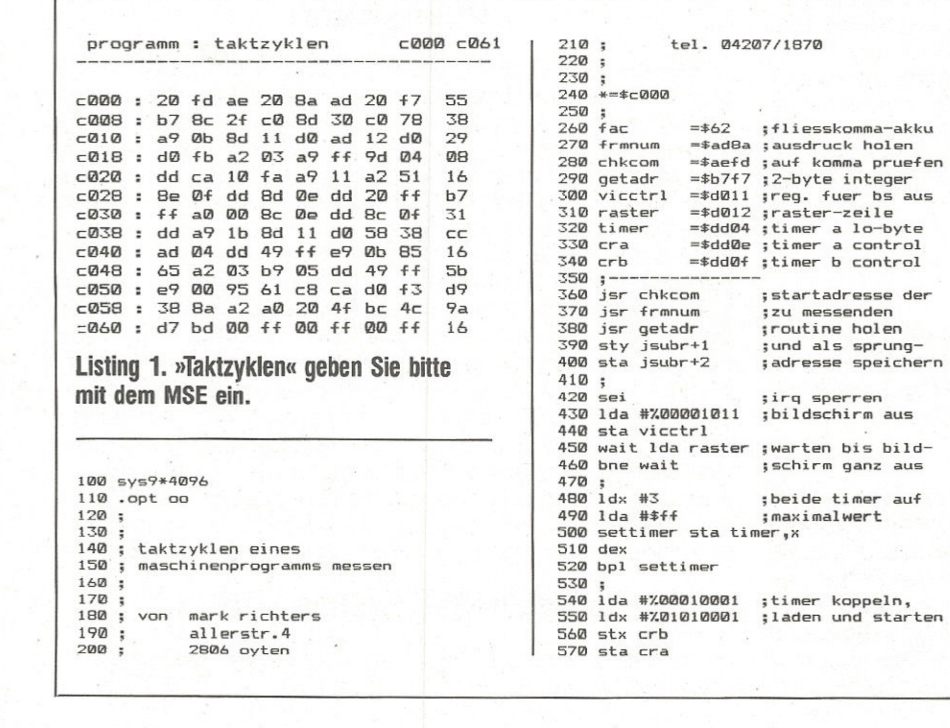


Bild 1. Flußdiagramm zum Laufzeit-Meß-System »LMS«. Nach diesem Schema berechnet man die Laufzeit eines Basic-Programms.



```
 programm : taktzyklen        c000 c061
----------------------------------------

c000 : 20 fd ae 20 8a ad 20 f7   55
c008 : b7 8c 2f c0 8d 30 c0 78   38
c010 : a9 0b 8d 11 d0 ad 12 d0   29
c018 : d0 fb a2 03 a9 ff 9d 04   08
c020 : dd ca 10 fa a9 11 a2 51   16
c028 : 8e 0f dd 8d 0e dd 20 ff   b7
c030 : ff a0 00 8c 0e dd 8c 0f   31
c038 : dd a9 1b 8d 11 d0 58 38   cc
c040 : ad 04 dd 49 ff e9 0b 85   16
c048 : 65 a2 03 b9 05 dd 49 ff   5b
c050 : e9 00 95 61 c8 ca d0 f3   d9
c058 : 38 8a a2 a0 20 4f bc 4c   9a
c060 : d7 bd 00 ff 00 ff 00 ff   16
```

Listing 1. »Taktzyklen« geben Sie bitte mit dem MSE ein.

```
100 sys9*4096
110 .opt oo
120 ;
130 ;
140 ; taktzyklen eines
150 ; maschinenprogramms messen
160 ;
170 ;
180 ; von   mark richters
190 ;       allerstr.4
200 ;       2806 oyten
210 ;       tel. 04207/1870
220 ;
230 ;
240 *=$c000
250 ;
260 fac       =$62   ;fliesskomma-akku
270 frmnum    =$ad8a ;ausdruck holen
280 chkcom    =$aefd ;auf komma pruefen
290 getadr    =$b7f7 ;2-byte integer
300 vicctrl   =$d011 ;reg. fuer bs aus
310 raster    =$d012 ;raster-zeile
320 timer     =$dd04 ;timer a lo-byte
330 cra       =$dd0e ;timer a control
340 crb       =$dd0f ;timer b control
350 ;---------------
360 jsr chkcom       ;startadresse der
370 jsr frmnum       ;zu messenden
380 jsr getadr       ;routine holen
390 sty jsubr+1      ;und als sprung-
400 sta jsubr+2      ;adresse speichern
410 ;
420 sei              ;irq sperren
430 lda #%00001011   ;bildschirm aus
440 sta vicctrl
450 wait lda raster ;warten bis bild-
460 bne wait         ;schirm ganz aus
470 ;
480 ldx #3           ;beide timer auf
490 lda #$ff         ;maximalwert
500 settimer sta timer,x
510 dex
520 bpl settimer
530 ;
540 lda #%00010001   ;timer koppeln,
550 ldx #%01010001   ;laden und starten
560 stx crb
570 sta cra
580 ;---------------
590 jsubr jsr $ffff ;sprung auf routine
600 ;---------------
610 ldy #0           ;beide timer stop
620 sty cra
630 sty crb
640 lda #%00011011   ;bildschirm wieder
650 sta vicctrl      ;an
660 cli              ;irq wieder frei-
670 ;
680 sec              ;4 timer-bytes
690 lda timer        ;umrechnen
700 eor #$ff         ;fuer aufruf be-
710 sbc #11          ;noetigte zyklen
720 sta fac+3        ;abziehen
730 ldx #3           ;und die restlichen
740 ;
750 l1 lda timer+1,y;3 timer-werte
760 eor #$ff         ;anpassen
770 sbc #0
780 sta fac-1,x
790 iny
800 dex
810 bne l1
820 ;
830 sec
840 txa              ;4 byte werten
850 ldx #$a0         ;ohne vorzeichen
860 jsr $bc4f        ;in fliesskommazahl
870 jmp $bdd7        ;fac ausgeben/rts

ready.
```

Listing 2. Information für Profis. Quell-Code-Listing zu »Taktzyklen«

```
100 REM ------------ LAUFZEIT-MESS-SYSTEM   <146>
110 REM --------------------LADEPROGRAMM   <109>
120 REM --------------------------------   <166>
130 REM -------------       FRANZ STOIBER   <150>
140 REM ------------- BRAEUNLICHGASSE 30   <036>
150 REM ------------- A-2700 WR.NEUSTADT   <194>
160 REM ---------------------- OESTEREICH   <197>
170 REM --------------------------------   <216>
180 IF L=1 THEN 260                        <045>
190 POKE 53280,15: POKE 53281,15: POKE 646
    ,6                                     <009>
200 GA=8                                   <033>
210 PRINT CHR$(14);"{CLR,11DOWN,6RIGHT}LAU
    FZEIT-MESS-SYSTEM{2SPACE}V2.0"         <122>
220 PRINT "{7RIGHT}(C) FRANZ STOIBER{2SPAC
    E}11/85"                               <237>
230 PRINT "{2DOWN,12RIGHT}BITTE WARTEN"    <096>
240 POKE 646,15                            <224>
250 L=1: LOAD "LMS.PHA",GA,1               <085>
260 SYS 40698                              <084>
270 PRINT "{HOME}";                        <202>
280 PRINT "{2DOWN}SYS 40780"               <137>
290 PRINT "{2DOWN}LOAD ";CHR$(34);"LMS.BAS
    ";CHR$(34);",";GA                      <189>
300 PRINT "{4DOWN}RUN"                     <234>
310 FOR I=631 TO 633: POKE I,13: NEXT      <226>
320 POKE 198,3                             <103>
330 PRINT "{HOME}";                        <006>
340 REM --------------------------------   <132>
350 REM FUER DAS ARBEITEN MIT DATASSETTE   <110>
360 REM SIND FOLGENDE ZEILEN ZU AENDERN:   <217>
370 REM                                    <178>
380 REM *** 200 GA=1                       <087>
390 REM *** 210 ENTFAELLT                  <174>
400 REM *** 220 ENTFAELLT                  <057>
410 REM *** 230 ENTFAELLT                  <195>
420 REM *** 240 POKE 646,0                 <247>
430 REM *** 270 PRINT "{CLR}";             <007>
440 REM *** 300 PRINT "{5DOWN}RUN"         <103>
450 REM                                    <002>
460 REM --------------------------------   <252>
470 REM AM BAND MUESSEN DIE PROGRAMME      <214>
480 REM IN DER REIHENFOLGE                 <215>
490 REM                                    <042>
500 REM LMS, LMS.PHA, LMS.BAS              <074>
510 REM                                    <062>
520 REM GESPEICHERT WERDEN.                <123>

© 64'er
```

Listing 3. Das Ladeprogramm »LMS« geben Sie bitte mit dem Checksummer V 3 (Ausgabe 3/86, Seite 55) ein.

Durch die Eingabe von END im READY-Modus wird das Meßprogramm interaktiv, und es verzweigt anschließend wieder zum Hauptmenü, in dem die Meßergebnisse des letzten Programmlaufs ausgewertet werden können.

Anzeige ab Nr. (a)

Diese Funktion gestattet die Eingabe einer Zeilennummer, ab der die Meßergebnisse am Bildschirm angezeigt werden.

Der Bildschirm zeigt die jeweilige Zeilennummer, die Anzahl der Aufrufe dieser Zeile und die Gesamtverweilzeit des Basic-Interpreters in dieser Zeile. Bei einer Aufrufanzahl größer als eins wird zusätzlich die durchschnittliche Laufzeit pro Aufruf angezeigt.

Die jeweils größte Laufzeit ist zur leichteren Orientierung durch »←« markiert.

Vorhergehende Seite (↑)

Durch Drücken der Taste (↑) wird die Ausgabe am Bildschirm um eine Seite zurückgeblättert.

Nächste Seite (RETURN)

Die Bildschirmausgabe wird mit »RETURN« um eine Seite weitergeblättert.

Summe (s)

Diese Funktion erlaubt die Berechnung der Programmlaufzeit bezogen auf einen einzugebenden Zeilenbereich.

Drucken (d)

Die Auswahl dieser Funktion bewirkt die Ausgabe der Meßergebnisse auf Drucker. Durch die Angabe von Zeilennummern kann der Ausgabebereich eingegrenzt werden.

Hilfe (h)

Das Menü wird angezeigt.

Ende (e)

Der Computer wird nach Bestätigung der Auswahl in den Einschaltzustand versetzt.

Wird bei der Eingabe einer Zeilennummer nur »RETURN« gedrückt, so wird die Zeilennummer ab inklusive 0 und die Zeilennummer bis inklusive 63999 angenommen.

Hinweise

Das Meßverfahren kann auf Programme mit bis zu 1022 Zeilen angewandt werden. Der Basic-Speicher ist auf zirka 30 KByte verkürzt. Die maximal erfaßbare Anzahl von Zeilenaufrufen beträgt 65535. Die maximal meßbare Laufzeit pro Zeile beträgt etwa 72 Minuten.

Wenn das LMS aktiviert ist, braucht es etwa 0,9 Millisekunden (ms) pro Zeilenaufruf. Dies führt zu einer Verlängerung der Programmlaufzeit während der Messung. Die Meßergebnisse sind jedoch Netto-Laufzeiten; das heißt die ange-

```
programm : lms.pha          9ce0 9ff4
--------------------------------------

9ce0 : 20 e1 9e a9 83 a2 9f 85  a4
9ce8 : f9 86 fa a2 53 20 c3 9e  ba
9cf0 : ad 08 03 8d dc 9f ad 09  a7
9cf8 : 03 8d dd 9f a9 0c 8d 08  6e
9d00 : 03 a9 9d 8d 09 03 20 67  e9
9d08 : 9f 4c 74 a4 20 e1 9e a5  56
9d10 : 9d f0 03 4c 7b 9d ad 0e  e7
9d18 : dd 29 fe 8d 0e dd ad 0f  a0
9d20 : dd 29 fe 8d 0f dd a5 39  ec
9d28 : cd d6 9f d0 07 a5 3a cd  84
9d30 : d7 9f f0 32 ad d7 9f c9  05
9d38 : ff f0 03 20 80 9e a5 39  7a
9d40 : 8d d6 9f a5 3a 8d d7 9f  84
9d48 : a9 ff 8d 04 dd 8d 05 dd  ef
9d50 : 8d 06 dd 8d 07 dd ad 0e  3c
9d58 : dd 29 d7 8d 0e dd ad 0f  16
9d60 : dd 29 d7 8d 0f dd 78 ad  96
9d68 : 0e dd 09 01 8d 0e dd ad  e3
9d70 : 0f dd 09 41 8d 0f dd 58  52
9d78 : 4c e4 a7 20 73 00 c9 80  84
9d80 : f0 0a c9 8a d0 03 20 b6  4c
9d88 : 9d 4c 3d 9e 20 67 9f ad  86
9d90 : dc 9f 8d 08 03 ad dd 9f  f5
9d98 : 8d 09 03 a9 52 8d 77 02  13
9da0 : a9 55 8d 78 02 a9 4e 8d  28
9da8 : 79 02 a9 0d 8d 7a 02 a9  36
9db0 : 04 85 c6 6c 02 03 20 43  f5
9db8 : 9e ad 02 03 8d da 9f ad  97
9dc0 : 03 03 8d db 9f a9 cf 8d  c5
9dc8 : 02 03 a9 9e 8d 03 03 a9  da
9dd0 : ff 8d d6 9f 8d d7 9f a9  a9
9dd8 : 00 85 fd a9 d0 85 fe a0  c6
9de0 : 00 b9 7b 9f 91 fd c8 c0  3d
9de8 : 08 d0 f6 a5 2b 85 fb a5  e5
9df0 : 2c 85 fc 20 b5 9e a0 00  f5
9df8 : b1 fb 8d d8 9f c8 b1 fb  24
9e00 : 8d d9 9f d0 12 ad d8 9f  ad
9e08 : d0 0d a9 ff a0 00 91 fd  15
9e10 : c8 91 fd 20 4d 9e 60 a0  b1
9e18 : 02 b1 fb a0 00 91 fd a0  cb
9e20 : 03 b1 fb a0 01 91 fd a9  f7
9e28 : 00 c8 91 fd c0 07 d0 f9  2c
9e30 : ad d8 9f 85 fb ad d9 9f  b6
9e38 : 85 fc 4c f3 9d 20 79 00  8e
9e40 : 4c e7 a7 48 78 a5 01 29  7e
9e48 : fc 85 01 68 60 48 a5 01  35
9e50 : 09 03 85 01 58 68 60 a9  fa
9e58 : 00 85 fd a9 d0 85 fe 20  45
9e60 : 43 9e d8 20 b5 9e a0 01  01
9e68 : b1 fd cd d7 9f d0 0c 88  48
9e70 : b1 fd cd d6 9f d0 04 20  3f
9e78 : 4d 9e 60 b0 fa 4c 63 9e  1f
9e80 : 20 57 9e 20 43 9e a0 02  a7
9e88 : 18 b1 fd 69 01 91 fd c8  4b
9e90 : b1 fd 69 00 91 fd 18 c8  95
9e98 : 20 4d 9e b9 00 dd 49 ff  52
9ea0 : 20 43 9e 71 fd 08 91 fd  9a
9ea8 : c0 07 f0 04 28 4c 97 9e  29
9eb0 : 28 20 4d 9e 60 18 a9 08  8d
9eb8 : 65 fd 85 fd a9 00 65 fe  6b
9ec0 : 85 fe 60 a0 00 b1 f9 20  a6
9ec8 : 16 e7 c8 ca d0 f7 60 20  ec
9ed0 : 80 9e ad da 9f 8d 02 03  da
9ed8 : ad db 9f 8d 03 03 6c 02  0a
9ee0 : 03 a9 02 8d 20 d0 60 20  34
9ee8 : 43 9e a0 00 b1 fd 99 de  d2
9ef0 : 9f c8 c0 08 d0 f6 20 4d  04
9ef8 : 9e 60 20 43 9e a9 00 8d  89
9f00 : 00 d0 20 4d 9e 60 20 fd  83
9f08 : ae 20 9e b7 8a 48 20 fd  cc
9f10 : ae 20 9e b7 68 a8 18 20  d9
9f18 : f0 ff 20 fd ae 4c a4 aa  05
9f20 : a9 00 8d 00 08 a9 01 85  0a
9f28 : 2b a9 08 85 2c a9 00 85  f6
9f30 : 37 a9 80 85 38 60 a9 00  3a
9f38 : 8d 00 80 a9 01 85 2b a9  57
9f40 : 80 85 2c a9 e0 85 37 a9  2d
9f48 : 9c 85 38 60 20 20 9f 20  83
9f50 : 44 a6 a2 00 b5 2b 9d e6  89
9f58 : 9f e8 e0 0e d0 f6 20 36  17
9f60 : 9f 20 44 a6 4c 74 a4 a2  35
9f68 : 00 bd e6 9f 48 b5 2b 9d  0e
9f70 : e6 9f 68 95 2b e8 e0 0e  8c
9f78 : d0 ef 60 4c 4d 53 2d 46  92
9f80 : 2e 53 2e 0d 4c 41 55 46  36
9f88 : 5a 45 49 54 2d 4d 45 53  5b
9f90 : 53 2d 53 59 53 54 45 4d  01
9f98 : 20 20 56 32 2e 30 20 20  c9
9fa0 : 31 31 2f 38 35 0d 28 43  20
9fa8 : 29 20 46 52 41 4e 5a 20  ed
9fb0 : 53 54 4f 49 42 45 52 0d  dc
9fb8 : 11 52 55 4e 20 2d 20 4d  98
9fc0 : 45 53 53 45 4e 0d 45 4e  2b
9fc8 : 44 20 2d 20 41 55 53 57  26
9fd0 : 45 52 54 45 4e 0d 00 00  49
9fd8 : 00 00 00 00 00 00 00 00  d9
9fe0 : 00 00 00 00 00 00 00 00  e1
9fe8 : 00 00 00 00 00 00 00 00  e9
9ff0 : 00 00 00 00 ff 00 ff 00  f0
```

Listing 4. Das Maschinenprogramm »LMS.PHA« bitte mit dem MSE eingeben.

zeigten Zeiten entsprechen den Laufzeiten ohne Meßprogramm. Es ist zu beachten, daß Programmschleifen innerhalb einer Basic-Zeile vom LMS nicht erkannt werden. Dies liegt an der zeilenorientierten Arbeitsweise des LMS.

```
10 GET A$: IF A$="" THEN 10
```

ergibt als Anzahl der Aufrufe der Zeile 10 die Zahl 1.

Um die exakte Aufrufanzahl zu erhalten, muß die Zeile geteilt werden.

```
10 GET A$
20 IF A$="" THEN 10
```

ergibt die genaue Anzahl der Aufrufe.

Die gleiche Vorgangsweise ist auch für Zeilen angebracht, in denen nach einem GOSUB noch weitere Statements folgen. In diesem Fall wird nämlich nach der Rückkehr vom Unterprogramm die Aufrufanzahl der Zeile um 1 erhöht. Die ausgewiesene durchschnittliche Laufzeit ist in diesem Fall mit Vorsicht zu genießen.

Im Rahmen von mehreren Messungen an einem Programm treten in der Regel unterschiedliche Laufzeiten für dieselben Statements auf. Diese zunächst etwas verblüffende Feststellung hat ihre Ursache darin, daß für die Abarbeitung eines Basic-Befehls nicht immer die gleiche Anzahl von Instruktionen durchlaufen wird. Dies ergibt sich zum Beispiel durch Bildschirm-Scrolling, Interrupt-Behandlung oder Garbage Collection. Bei der Festlegung der Meßgenauigkeit wurde diese Tatsache berücksichtigt; das heißt nicht aussagekräftige Dezimalstellen wurden weggelassen.

Das LMS funktioniert nicht mit Programmen
— die den vom LMS belegten Speicher benötigen ($8000 bis $9FFF und $D000 bis $FFFF; siehe auch Programmbeschreibung),
— die Basic-Zeilen generieren oder
— die den NMI-Timer verwenden (zum Beispiel RS232-Programme).

Programmbeschreibung

Das Laufzeit-Meß-System besteht aus dem Maschinenprogramm »LMS.PHA« und den Basic-Programmen »LMS« und »LMS.BAS.«. »LMS.PHA« ist das eigentliche Meßprogramm. Es liegt im Speicherbereich $9CE0 bis $9FFF und wird bei seiner Aktivierung in die Basic-Interpreterschleife eingebunden. Für die Zeitmessung wird der Timer im CIA2 benutzt. Als Zeitbasis wird der Systemtakt verwendet. Die Meßergebnisse werden im RAM ab $D000 unter dem Betriebssystem abgelegt.

Der Ablauf einer Messung kann dem Flußdiagramm (siehe Bild 1) beziehungsweise dem Listing »LMS.ASS« entnommen werden. Die Aufbereitung und Auswertung der Meßergebnisse erfolgt im Programm »LMS.BAS«. Um ein abwechselndes Laden des in Basic geschriebenen Auswerteprogramms und des zu messenden Programms zu vermeiden, wird der Basic-Speicher derart geteilt, daß beide Programme gleichzeitig geladen sein können. LMS.BAS muß sich dabei mit weniger als 8 KByte ab Adresse $8000 begnügen.

Die Umschaltung des Basic-Speichers erfolgt in »LMS.PHA«. Beim Aktivieren des Meßprogramms wird das zu messende Programm zur Verfügung gestellt, beim Desaktivieren wird zum Auswerteprogrammm verzweigt.

Das Ladeprogramm »LMS« lädt zunächst »LMS.PHA«, damit die Maschinenroutinen zur Verfügung stehen. Danach teilt und initialisiert das Programm den Basic-Bereich. Zuletzt lädt und startet es das Auswerteprogramm.

Die Anpassung von »LMS.BAS« an unterschiedliche Drucker ist problemlos, da außer der Sekundäradresse 7 keine speziellen Steuercodes verwendet wurden.

Für die Verwendung mit Datassette müssen die im Programmlisting LMS enthaltenen Hinweise beachtet werden.

(Mark Richters/Franz Stoiber/og)

```
100 REM ------ LMS-AUSWERTEPROGRAMM V2.0   <031>
101 REM ------- 11/85      FRANZ STOIBER   <181>
110 PRINT CHR$(14);CHR$(8): POKE 808,225:
    ZF=6                                   <072>
120 E$="****************** ENDE ***********
    ******"                                <058>
130 POKE 53280,15: POKE 53281,15: POKE 646
    ,ZF: GOSUB 1440                        <240>
140 BL$="{40SPACE}"                        <040>
150 RA=53248: PU=253: L=22: CL=985.248: R=
    .05                                    <086>
160 K0=2↑8: K1=2↑16: K2=2↑24: K3=2↑32: K4=
    K1-1: K5=63999: K6=10                  <076>
170 RD=40679: AT=40710: AB=40535: NR=40918 <128>
180 P1=40926: P2=P1+1: P3=P1+2: P4=P1+3: P
    5=P1+4: P6=P1+5: P7=P1+6: P8=P1+7      <044>
200 J=0: K=80: P=RA                        <194>
210 KZ$="{6SPACE}NR{3SPACE}ANZ{8SPACE}MSEK
    {4SPACE}MSEK/ANZ"                      <218>
220 SYS AT,6,5,"MESSUNG ...............{3SP
    ACE}M"                                 <103>
230 SYS AT,6,7,"ANZEIGE AB NR ........{3SP
    ACE}A"                                 <155>
240 SYS AT,6,9,"VORHERGEHENDE SEITE ..{3SP
    ACE}↑"                                 <194>
250 SYS AT,6,11,"NAECHSTE SEITE ....... RE
    TURN"                                  <111>
260 SYS AT,6,13,"SUMME .................{3S
    PACE}S"                                <154>
270 SYS AT,6,15,"DRUCKEN ...............{3S
    PACE}D"                                <235>
280 SYS AT,6,17,"HILFE .................{3S
    PACE}H"                                <171>
290 SYS AT,6,19,"ENDE ..................{3S
    PACE}E"                                <137>
300 POKE 198,0: SYS AT,0,24,"{RVSON,4SPACE
    }WEITER MIT: M/A/↑/RETURN/S/D/H/E{3SPA
    CE,RVOFF}";                            <144>
310 POKE 2023,160: POKE 56295,ZF           <206>
320 GET A$: IF A$="" THEN 320              <131>
330 IF A$="M" THEN 430                     <039>
340 IF A$="A" THEN 450                     <059>
350 IF A$="↑" THEN 490                     <244>
360 IF A$=CHR$(13) THEN 540                <156>
370 IF A$="S" THEN 570                     <118>
380 IF A$="D" THEN 720                     <217>
390 IF A$="H" THEN 1010                    <211>
400 IF A$="E" THEN 1030                    <123>
410 GOTO 300                               <100>
420 REM ------------------------ MESSUNG   <090>
430 POKE 808,237: PRINT CHR$(9);"{CLR}";:
    SYS 40160                              <057>
440 REM --------------------- ANZEIGE AB   <208>
450 GOSUB 1930: IF F THEN 300              <252>
460 GOSUB 1440: GOSUB 1480                 <002>
470 P=P-8: GOSUB 1130: GOTO 300            <116>
480 REM ------------------------ ZURUECK   <085>
490 GOSUB 1930: IF F THEN 300              <036>
500 P=P-(ZZ+L-1)*8                         <057>
510 IF P<RA THEN P=RA                      <096>
520 GOSUB 1130: GOTO 300                   <157>
530 REM ---------------------------- VOR   <145>
540 GOSUB 1930: IF F THEN 300              <088>
550 GOSUB 1130: GOTO 300                   <187>
560 REM -------------------------- SUMME   <221>
570 GOSUB 1930: IF F THEN 300              <118>
580 GOSUB 1440                             <220>
590 SYS AT,5,4,"SUMME": SYS AT,5,5,"TTTTT" <170>
600 GOSUB 1480: GOSUB 1580                 <021>
610 P=P-8: S=0                             <089>
620 GOSUB 1080                             <069>
630 IF Z>ZB THEN 670                       <167>
640 GOSUB 1370                             <104>
650 S=S+T: SYS AT,30,10,Z                  <203>
660 GOTO 620                               <152>
670 SYS AT,30,10,"{6SPACE}"                <233>
680 T=S: T$="": GOSUB 1400                 <215>
690 SYS AT,5,17,"SUMME =";T$;" MSEK"       <243>
700 P=RA: GOTO 300                         <136>
710 REM ------------------------ DRUCKEN   <067>
720 GOSUB 1930: IF F THEN 300              <012>
730 GOSUB 1440                             <114>
740 SYS AT,5,4,"DRUCKEN": SYS AT,5,5,"TTTT
    TTT"                                   <085>
750 GOSUB 1480: GOSUB 1580: P=P-8: S=0     <078>
760 SYS AT,5,15,"DRUCKER O.K. ?"           <216>
770 SYS AT,5,16,"WEITER MIT BELIEBIGER TAS
    TE"                                    <232>
780 POKE 198,0: WAIT 198,1: POKE 198,0     <121>
790 OPEN 4,4,7: POKE 768,185: PRINT#4,"";:
```

```
     POKE 768,139                                <130>
 800 IF ST THEN SYS AT,8,22,"{RVSON,SPACE}D
     RUCKER NICHT BEREIT{SPACE,RVOFF}": CLO
     SE 4: GOTO 300                              <197>
 810 IF K<80 THEN PL$="": GOSUB 1830: GOTO
     830                                         <057>
 820 SYS AT,0,15,BL$;: SYS AT,0,16,BL$;          <176>
 830 F=0: PL$=""                                 <152>
 840 GOSUB 1080                                  <035>
 850 IF Z>ZB THEN 960                            <134>
 860 GOSUB 1320                                  <166>
 870 GOSUB 1370                                  <080>
 880 S=S+T: T$="": B$="{12SPACE}"                <020>
 890 GOSUB 1400                                  <148>
 900 IF AZ<2 THEN 920                            <153>
 910 B$="": T=T/AZ: AZ=0: GOTO 890               <043>
 920 PL$=PL$+"{2SPACE}"+Z$+AZ$+T$+B$             <010>
 930 IF F=0 THEN F=1: GOTO 840                   <162>
 940 GOSUB 1830                                  <102>
 950 GOTO 830                                    <220>
 960 IF F THEN GOSUB 1830                        <076>
 970 T=S: T$="": GOSUB 1400                      <251>
 980 PL$="{7SPACE}SUMME ="+T$+" MSEK": GOSU
     B 1830                                      <253>
 990 CLOSE 4: P=RA: GOTO 300                     <006>
1000 REM -------------------------- HILFE        <177>
1010 GOSUB 1440: GOTO 220                        <223>
1020 REM ---------------------------- ENDE       <017>
1030 GOSUB 1440: SYS AT,5,13,"BITTE ENDE B
     ESTAETIGEN (J/N):{SPACE,RVSON,SPACE,R
     VOFF}";                                     <245>
1040 GET A$: IF A$="" THEN 1040                  <203>
1050 IF A$="J" THEN SYS AT,35,13,"J": SYS
     64738                                       <092>
1060 SYS AT,0,13,BL$;: GOTO 220                  <094>
1070 REM ---------- WERTE AUS RAM LESEN          <066>
1080 P=P+8: PH=INT(P/K0): PL=P-PH*K0             <108>
1090 POKE PU,PL: POKE PU+1,PH: SYS RD            <161>
1100 Z=PEEK(P1)+PEEK(P2)*K0                      <143>
1110 RETURN                                      <152>
1120 REM ----------- AUSGABE BILDSCHIRM          <242>
1130 PRINT "{CLR,RVSON,4SPACE}NR{3SPACE}AN
     Z{8SPACE}MSEK{4SPACE}MSEK/ANZ{4SPACE}
     "                                           <212>
1140 M1=0: M2=0: C1$=" ": C2$=" "                <113>
1150 FOR ZZ=1 TO L                               <067>
1160 :GOSUB 1080                                 <035>
1170 :IF Z=K4 THEN P=P-8: PRINT E$;: GOTO
     1280                                        <083>
1180 :GOSUB 1320                                 <021>
1190 :GOSUB 1370                                 <096>
1200 :IF T>M1 THEN M1=T: L1=ZZ: C1$="←"          <247>
1210 :T$=""                                      <125>
1220 :GOSUB 1400                                 <221>
1230 :IF AZ<2 THEN 1260                          <003>
1240 :T=T/AZ: IF T>M2 THEN M2=T: L2=ZZ: C2
     $="←"                                       <048>
1250 :AZ=0: GOTO 1220                            <071>
1260 :PRINT Z$;AZ$;T$                            <177>
1270 NEXT                                        <008>
1280 IF C1$="←" THEN SYS AT,24,L1+1,C1$;         <196>
1290 IF C2$="←" THEN SYS AT,36,L2+1,C2$;         <105>
1300 RETURN                                      <088>
1310 REM ---------- AUFBEREITUNG NR,ANZ          <022>
1320 Z$=RIGHT$("{5SPACE}"+STR$(Z),6)             <031>
1330 AZ=PEEK(P3)+PEEK(P4)*K0                     <060>
1340 AZ$=RIGHT$("{5SPACE}"+STR$(AZ),6)           <129>
1350 RETURN                                      <138>
1360 REM --------------- ZEIT ERRECHNEN          <081>
1370 T=(PEEK(P5)+PEEK(P6)*K0+PEEK(P7)*K1+P
     EEK(P8)*K2)/CL                              <035>
1380 RETURN                                      <168>
1390 REM ------------ AUFBEREITUNG ZEIT          <076>
1400 TR=T+R: TH=INT(TR): TL=INT((TR-TH)*K6
     )                                           <187>
1410 T$=T$+RIGHT$("{9SPACE}"+STR$(TH),10)+
     "."+RIGHT$(STR$(TL),1)                      <016>
1420 RETURN                                      <208>
1430 REM ------------------- KOPFZEILEN          <051>
1440 PRINT "{CLR,RVSON,2SPACE}LAUFZEIT{2SP
     ACE}-{2SPACE}MESS{2SPACE}-{2SPACE}SYS
     TEM{3SPACE}V2.0{3SPACE}";                   <017>
1450 PRINT "{RVSON,2SPACE}(C) 11/85{SPACE,
     SHIFT-SPACE}FRANZ STOIBER{14SPACE}"         <252>
1460 RETURN                                      <248>
1470 REM ---------- EINGABE ZEILENNR AB          <039>
1480 RO=8: SYS AT,5,RO,"AB NR{2SPACE}?{2SP
     ACE}";: GOSUB 1650                          <083>
1490 IF A$="{5SPACE}" THEN ZA=0: SYS AT,14
     ,RO,ZA: GOTO 1520                           <062>
1500 ZA=VAL(A$): IF ZA<=K5 THEN SYS AT,0,2
     2,BL$;: GOTO 1520                           <169>
1510 SYS AT,6,22,"{RVSON,4SPACE}ZEILENNUMM
     ER ZU GROSS{4SPACE,RVOFF}";: GOTO 148
     0                                           <231>
1520 ZH=INT(ZA/K0): ZL=ZA-ZH*K0: POKE NR,Z
     L: POKE NR+1,ZH                             <064>
1530 SYS AB: P=PEEK(PU)+PEEK(PU+1)*K0            <070>
1540 P=P-8: GOSUB 1080                           <254>
1550 IF Z<K4 THEN RETURN                         <044>
1560 SYS AT,6,22,"{RVSON,SPACE}ZEILENNUMME
     R NICHT GEFUNDEN{SPACE,RVOFF}";: GOTO
      1480                                       <208>
1570 REM --------- EINGABE ZEILENNR BIS          <165>
1580 RO=10: SYS AT,5,RO,"BIS NR ?{2SPACE}"
     ;: GOSUB 1650                               <138>
1590 IF A$="{5SPACE}" THEN ZB=K5: SYS AT,1
     4,RO,ZB: RETURN                             <162>
1600 ZB=VAL(A$): IF ZB>=ZA THEN 1620             <120>
1610 SYS AT,5,22,"{RVSON,2SPACE}ZEILENNR B
     IS < ZEILENNR AB{2SPACE,RVOFF}";: GOT
     O 1580                                      <118>
1620 IF ZB>K5 THEN ZB=K5                         <121>
1630 SYS AT,0,22,BL$;: RETURN                    <243>
1640 REM ---------------- ZAHL EINGEBEN          <217>
1650 N=1: RC=1024+14+40*RO: CO=RC+54272: P
     OKE 198,0                                   <012>
1660 FOR I=1 TO 5: POKE CO+I,ZF: NEXT            <171>
1670 IF N>5 THEN N=5                             <031>
1680 IF N<1 THEN N=1                             <193>
1690 PO=RC+N: POKE PO,PEEK(PO)+128               <040>
1700 GET C$: IF C$="" THEN 1700                  <051>
1710 IF C$>="0" AND C$<="9" THEN POKE PO,A
     SC(C$): N=N+1: GOTO 1670                    <239>
1720 IF C$=CHR$(157) THEN POKE PO,PEEK(PO)
     -128: N=N-1: GOTO 1670                      <171>
1730 IF C$=CHR$(29) THEN POKE PO,PEEK(PO)-
     128: N=N+1: GOTO 1670                       <120>
1740 IF C$=CHR$(20) THEN SYS AT,15,RO,"{5S
     PACE}";: GOTO 1650                          <206>
1750 IF C$=CHR$(13) THEN POKE PO,PEEK(PO)-
     128: GOTO 1770                              <164>
1760 GOTO 1700                                   <032>
1770 A$=""                                       <019>
1780 FOR I=1 TO 5                                <100>
1790 :A$=A$+CHR$(PEEK(RC+I))                     <021>
1800 NEXT                                        <032>
1810 RETURN                                      <090>
1820 REM ---------------- ZEILE DRUCKEN          <170>
1830 IF K<68 THEN 1900                           <130>
1840 IF K<73 THEN PRINT# 4,"": K=K+1: GOTO
      1840                                       <038>
1850 J=J+1: PRINT# 4,"": PRINT# 4,""             <202>
1860 PRINT# 4,"{3SPACE}LAUFZEIT - MESS - S
     YSTEM{2SPACE}V2.0";                         <208>
1870 PRINT# 4,"{5SPACE}(C) FRANZ STOIBER 1
     1/85{6SPACE}SEITE: ";RIGHT$(STR$(J+10
     0),2)                                       <115>
1880 PRINT# 4,"": PRINT# 4,"": PRINT# 4,KZ
     $;KZ$: PRINT# 4,""                          <144>
1890 K=8                                         <181>
1900 PRINT# 4,PL$: K=K+1                         <050>
1910 RETURN                                      <190>
1920 REM ------------- TEST AUF MESSUNG          <159>
1930 PP=P: F=0: P=RA-8: A$=""                    <032>
1940 GOSUB 1080                                  <119>
1950 FOR I=P1 TO P8: A$=A$+CHR$(PEEK(I)):
     NEXT                                        <050>
1960 IF A$="LMS-F.S." THEN 1990                  <174>
1970 SYS AT,6,22,"{RVSON,SPACE}KEINE MESSU
     NG GELAUFEN !{SPACE,RVOFF}"                 <019>
1980 F=1: RETURN                                 <117>
1990 GOSUB 1080                                  <169>
2000 IF Z<K4 THEN P=PP: RETURN                   <078>
2010 SYS AT,6,21,"{RVSON,SPACE}WAEHREND DE
     R MESSUNG WAR "                             <057>
2020 SYS AT,6,22,"{RVSON,2SPACE}KEIN PROGR
     AMM GELADEN!{2SPACE,RVOFF}"                 <083>
2030 GOTO 1980                                   <081>
```


Listing 5. Das Auswertungsprogramm »LMS.BAS« (Schluß)

Tips und Tricks zum EPROMer

Zum EPROM-Brenner aus unseren Ausgaben 12/85 und 1/86 haben sich einige Fragen ergeben. Hier finden Sie, welche Probleme entstanden sind und wie Sie diese lösen können.

Beim 64'er EPROM-Brenner haben sich leider durch die kontinuierliche Weiterentwicklung Probleme ergeben. Wir bedauern, daß dadurch Schwierigkeiten beim Nachbau entstanden sind. Welche Korrekturen für eine einwandfreie Funktion des EPROM-Brenners notwendig sind, erfahren Sie hier.

— Wenn Sie die Platine mit Hilfe des abgedruckten Layouts (Ausgabe 1/86, Seite 149) selbst gefertigt haben, müssen Sie die Verbindung zwischen Pin 8 und 19 des Textool-Sockels beseitigen. Sie entsteht durch einen Kurzschluß neben Pin 1 bis Pin 4 bei IC 1.

— Im Bestückungsplan (Ausgabe 1/86, Seite 151) sind die Zehnerdioden ZD 2 bis ZD 7 verkehrt gepolt und die Diode D 7 fehlt. Deshalb haben wir einen neuen Bestückungsplan (Bild) abgedruckt. Dabei sind die vorher unter R 6 zusammengefaßten Widerstände aufgeteilt in R 6a und R 6b.

— In der Stückliste (Ausgabe 1/86, Seite 151) sind Änderungen bei R 6 und R 8 notwendig. Eine neue Stückliste (Tabelle) ist daher ebenfalls abgedruckt.

— In der Kaskadenschaltung (Ausgabe 12/85, Seite 44) fehlt der Kondensator zwischen den beiden mittleren Dioden in der oberen Leitung. Die Polung ist identisch mit allen anderen Kondensatoren.

— Nun kommen wir zum Schaltplan (Ausgabe 12/85, Seite 47). Vor den Zehnerdioden ZPD 24 (ZD 2 und ZD 3), ZPD 9,1 + ZPD 3,3 (ZD 4 bis ZD 7) sind jeweils Vorwiderstände (R 6) mit einem Wert von 63 Ohm eingezeichnet. Diese Widerstände müssen durch die neuen Widerstände R 6a und R 6b ersetzt werden. Die 43 Ohm-Widerstände sind vor die beiden ZPD 24 zu schalten, während die zwei weiteren gegen 430 Ohm-Widerstände ausgetauscht werden.

Zwischen Pin 7 des IC 5 (74 LS 139) und T 5 befinden sich je ein Gatter von IC 4 (7404) und IC 3 (7406). Zwischen diesen beiden Gattern muß die Anode einer nicht eingezeichneten Diode (D 7) angeschlossen sein. Die Kathode liegt an PS 1.

Die Platine für den EPROM-Brenner und den Spannungswandler können Sie auch über unseren Hardware-Service erhalten. (kn)

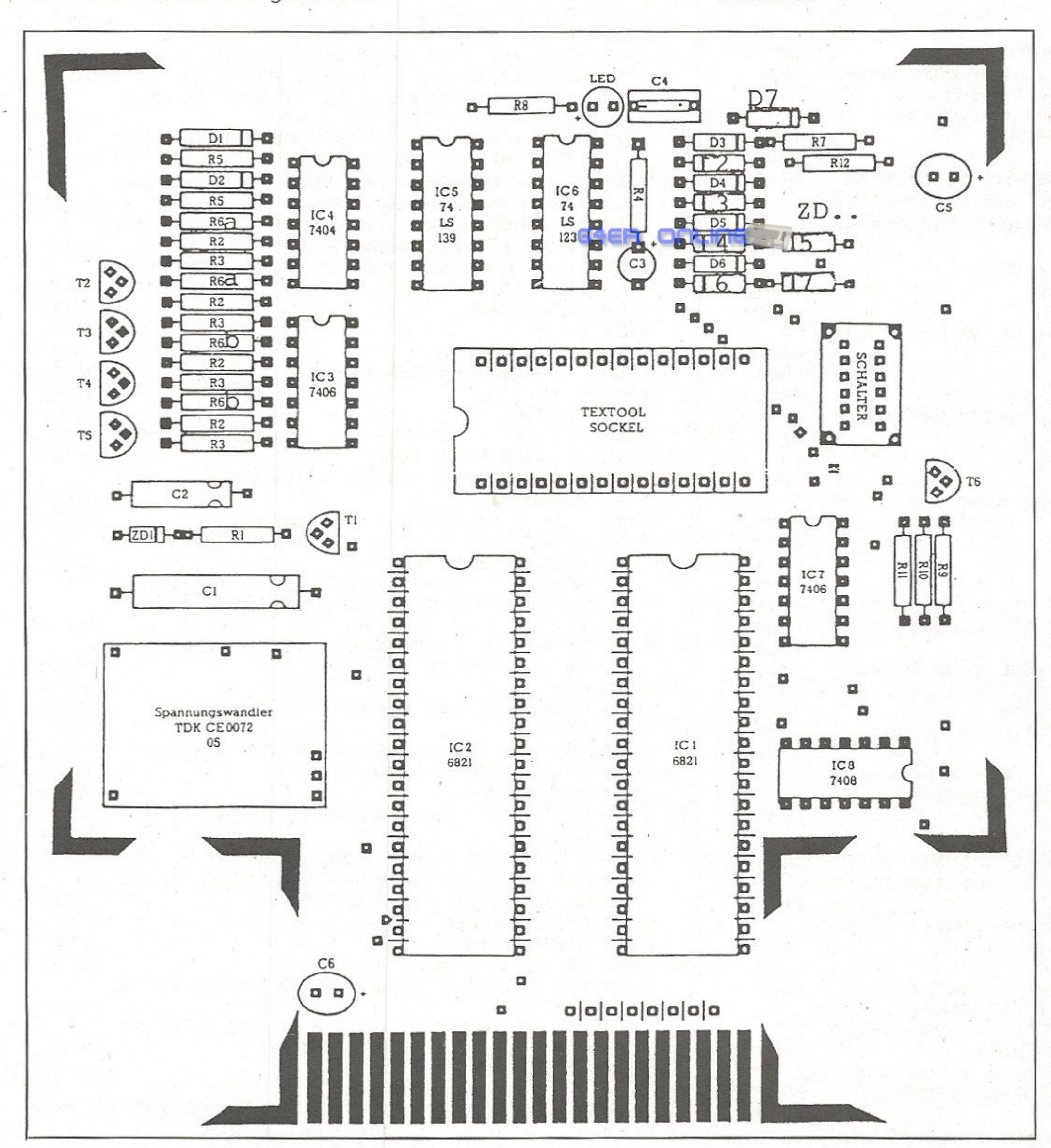


Bild. Bestückungsplan des EPROM-Brenners

Halbleiter

IC 1/2	6821
IC 3/7	7406
IC 4	7404
IC 5	74 LS 139
IC 6	74 LS 123
IC 8	7408
T 1	BC 337-40
T 2 - T 6	BC 327-40
D 1 - D 7	1 N 4148
ZD 1	ZPD 27
ZD 2/3	ZPD 24
ZD 4/6	ZPD 9.1
ZD 5/7	ZPD 3.3
LD 1	LED ROT 3 mm

Kondensatoren

C 1	100 µF/ 40V Elko
C 2	10 µF/ 40V Elko
C 3	6.8 µF/ 10V Tantal
C 4	100 µF/100V MKT
C 5/6	10 µF/ 10V Tantal

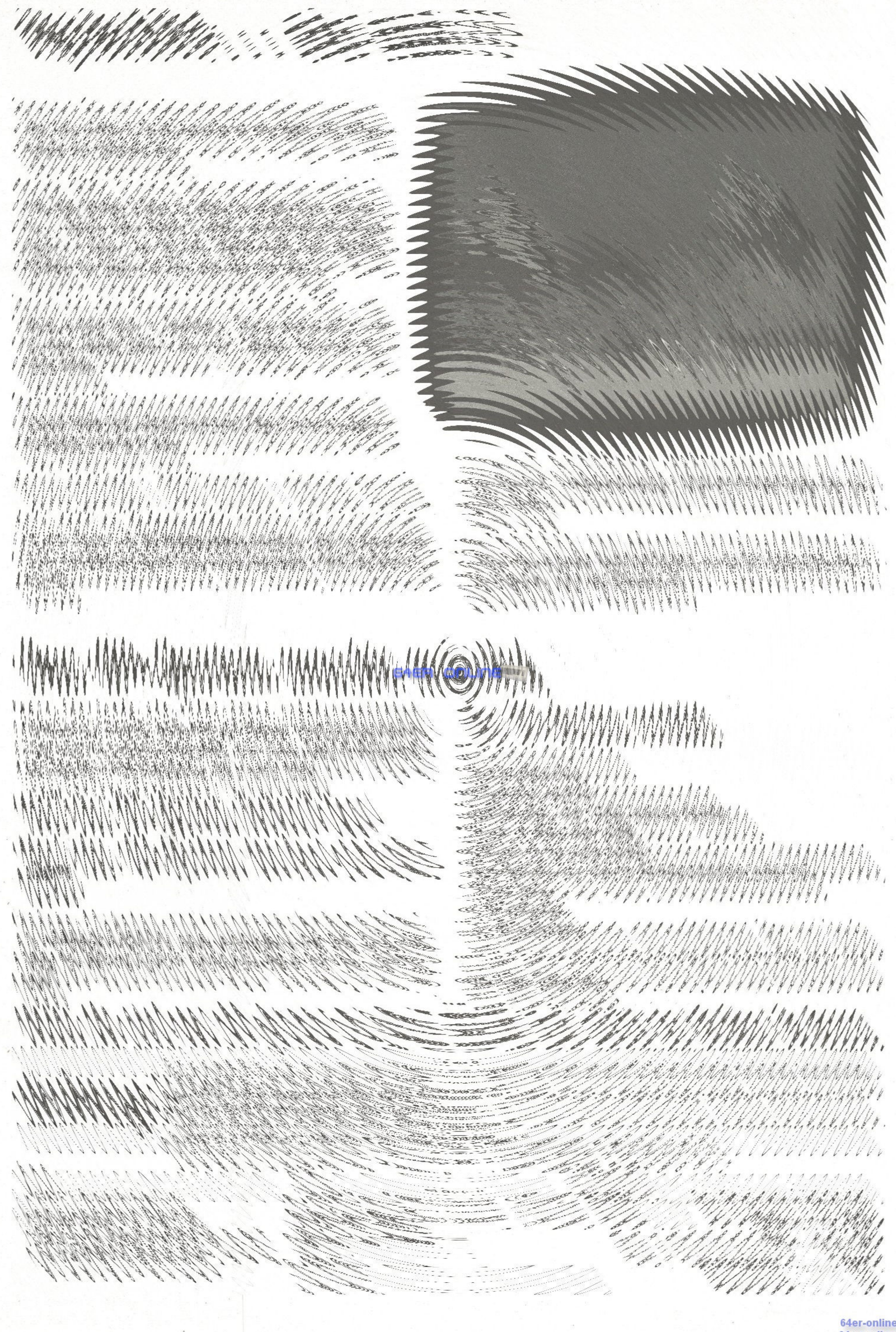
Widerstände

R1	1 KOhm	
R2	3.3 KOhm	4 Stück
R3	1 KOhm	4 Stück
R4	20.5 KOhm	Metallf.1%
R5	1 KOhm	2 Stück
R6 a	43 Ohm	2 Stück
R6 b	430 Ohm	2 Stück
R7	4.7 KOhm	
R8	220 Ohm	
R9	3.3 KOhm	
R10	1 KOhm	
R11	4.7 KOhm	
R12	10 KOhm	

Sonstige Bauteile

1 Textool-Sockel 28polig
1 Spannungswandler TDK CE 0072
1 Vierfach Umschalter Typ SL 422 P (wie Bürklin 11 G 822 oder Conrad 708232)

Stückliste für den EPROM-Brenner

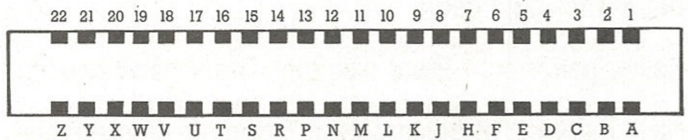




64'er Extra

Das 64'er Extra bringt geballte Informationen über Ihren C 64 zum Heraustrennen und Sammeln. In dieser achten Ausgabe finden Sie eine Übersicht zum Expansion-Port. Damit Sie nicht mehr die verschiedensten Bücher gleichzeitig wälzen müssen, haben wir für Sie in Klartext die Funktionen aller Leitungen des Expansion-Ports beschrieben.

Am Expansion-Port sind alle Leitungen herausgeführt, die für eine hardwaremäßige Erweiterung Ihres Computers notwendig sind. In den meisten Unterlagen findet man allerdings Angaben, die Ihnen bei der Durchdringung des Signaldschungels nicht entscheidend weiterhelfen. Mit unserer achten Ausgabe des 64'er Extra wollen wir Ihnen eine kompakte aber dennoch ausführliche Information über den Expansion-Port zur Verfügung stellen. Keine Leitung des Expansion-Ports wird ausgelassen. Die Abbildung zeigt den Expansion-Port, wie er von hinten betrachtet am Computer zu sehen ist. Das heißt die Zahlen gelten für die obere Kontaktleiste und beginnen von hinten gesehen rechts mit 1. In der Tabelle finden Sie häufig einen Strich über einer Signalbezeichnung. Hier handelt es sich um sogenannte »Nicht-Anweisungen«, die auch als Low-aktiv bezeichnet werden. Das bedeutet, wenn diese Leitung auf 0 (Low) gelegt wird, ist die betreffende Funktion aktiv.



Der Expansion-Port des C 64 — von hinten gesehen

Der Expansion-Port des C 64

Pin	Name	Beschreibung	Bemerkung	Eingang/ Ausgang
1	GND	Systemmasse		A
2	+5 V DC	Betriebsspannung	Belastung maximal 450 mA	A
3	+5 V DC			A
4	IRQ	Interrupt-Anforderung	Wenn IRQ auf 0 gesetzt wird, arbeitet der Prozessor den momentanen Befehl ab, rettet den Programmzähler und das Statusregister auf den Stapel und lädt den Programmzähler mit den Speicherzellen $FFFE und $FFFF. Diese Speicherzellen enthalten die Einsprungadresse für die IRQ-Routine des Betriebssystems ($FF48).	E
5	R/W	Lesen/Schreiben	Der Prozessor setzt diese Leitung bei jedem Lesezyklus (beispielsweise LDA) auf 1 und bei jedem Schreibzyklus (beispielsweise STA) auf 0.	A
6	DOT CLOCK	Taktfrequenz des Video Interface Controller (VIC)	7,80 MHz für PAL	A
7	I/O1	Ein-/Ausgabe-Bereich 1 ($DE00-$DEFF)	Ist die Speicherzelle 1 (Zeropage) auf $37 gesetzt und es wird eine Adresse zwischen $DE00 und $DEFF angesprochen, so geht diese Leitung auf 0 (Chip Select für externe Peripheriebausteine, ungepuffert).	A
8	GAME	Einblendung externer Bausteine Bereich ($A000 — $BFFF)	Wird GAME auf 0 gelegt, schaltet der Computer den Basic-Interpreter ab und erzeugt für den Bereich von $A000 bis $BFFF ein Chip Select. Wird nun eine Adresse in diesem Bereich angesprochen, so geht die Leitung ROMH (Pin B) auf 0.	E

Pin	Name	Beschreibung	Bemerkung	Eingang/ Ausgang
9	EXROM	Einblendung externer Bausteine Bereich ($8000 — $9FFF)	Wird EXROM auf 0 gelegt, schaltet der Computer das RAM für den Bereich $8000 bis $9FFF ab und erzeugt ein Chip Select. Wird nun eine Adresse in diesem Bereich angesprochen, geht die Leitung ROML (Pin 11) auf 0.	E
10	I/O2	Ein-/Ausgabe-Bereich 2 ($DF00 — $DFFF)	Ist die Speicherzelle 1 (Zeropage) auf $37 gesetzt und eine Adresse zwischen $DF00 und $DFFF wird angesprochen, so geht diese Leitung auf 0 (Chip Select für externe Peripheriebausteine, gepuffert)	A
11	ROML	Chip Select für den Bereich I (EXROM)	siehe Pin 9 (EXROM)	A
12	BA	Buszugriff vom VIC	BA liegt auf 1, wenn der VIC auf den Daten- oder Adreßbus zugreift. Ist BA = 1, so dürfen keine externen Bausteine auf den Bus zugreifen.	A
13	DMA	Direkter Zugriff von externen Bausteinen auf internes RAM	Liegt DMA auf 0, so hält der Prozessor nach dem momentanen Lesezyklus an, damit der externe Baustein auf den Bus zugreifen kann. Wenn DMA während eines Schreibzyklus auf 0 gesetzt wird, ignoriert der Prozessor dies bis zum nächsten Lesezyklus. Ist die Leitung wieder hochohmig, setzt der Prozessor seine Arbeit fort.	E
14	D7	Datenbus Bit 7		
15	D6	Datenbus Bit 6		
16	D5	Datenbus Bit 5		
17	D4	Datenbus Bit 4	Der Datenbus ist nicht gepuffert	E/A
18	D3	Datenbus Bit 3		
19	D2	Datenbus Bit 2		
20	D1	Datenbus Bit 1		
21	D0	Datenbus Bit 0		
22	GND	Systemmasse		A
A	GND	Systemmasse		
B	ROMH	Chip Select für Bereich II (GAME)	siehe Pin 8 (GAME)	A
C	RESET	Computer-Kaltstart	Liegt RESET auf 0, so werden Stapel und Statusregister neu initialisiert. Der Programmzähler wird mit dem Inhalt der Speicherzellen $FFFC und $FFFD geladen. Die Speicherzellen enthalten die Einsprungadresse für die RESET-Routine des Betriebssystems ($FCE2).	A
D	NMI	Nicht maskierbarer Interrupt (Computer-Warmstart, identisch mit RUN/STOP-RESTORE)	Der Prozessor erkennt diese Interrupt-Anforderung mit fallender Flanke auf dieser Leitung, das heißt mit einem Übergang von 1 auf 0. Der Befehl wird abgearbeitet und der Programmzähler sowie das Statusregister auf den Stapel gerettet. Anschließend wird der Programmzähler mit dem Inhalt der Speicherzellen $FFFA und $FFFB geladen. Diese Speicherzellen enthalten die Einsprungadresse für die NMI-Routine des Betriebssystems ($FE43).	E
E	Φ 2	Systemtakt (0,98 MHz)	Bei fallender Flanke von 2 sind Daten gültig	A
F	A 15	Adreß-Bus Bit 15		
H	A 14	Adreß-Bus Bit 14		
J	A 13	Adreß-Bus Bit 13		
K	A 12	Adreß-Bus Bit 12		
L	A 11	Adreß-Bus Bit 11		
M	A 10	Adreß-Bus Bit 10		
N	A 9	Adreß-Bus Bit 9		
P	A 8	Adreß-Bus Bit 8	Der Adreß-Bus ist nicht gepuffert	E/A
R	A 7	Adreß-Bus Bit 7		
S	A 6	Adreß-Bus Bit 6		
T	A 5	Adreß-Bus Bit 5		
U	A 4	Adreß-Bus Bit 4		
V	A 3	Adreß-Bus Bit 3		
W	A 2	Adreß-Bus Bit 2		
X	A 1	Adreß-Bus Bit 1		
Y	A 0	Adreß-Bus Bit 0		
Z	GND	Systemmasse		A

64'er

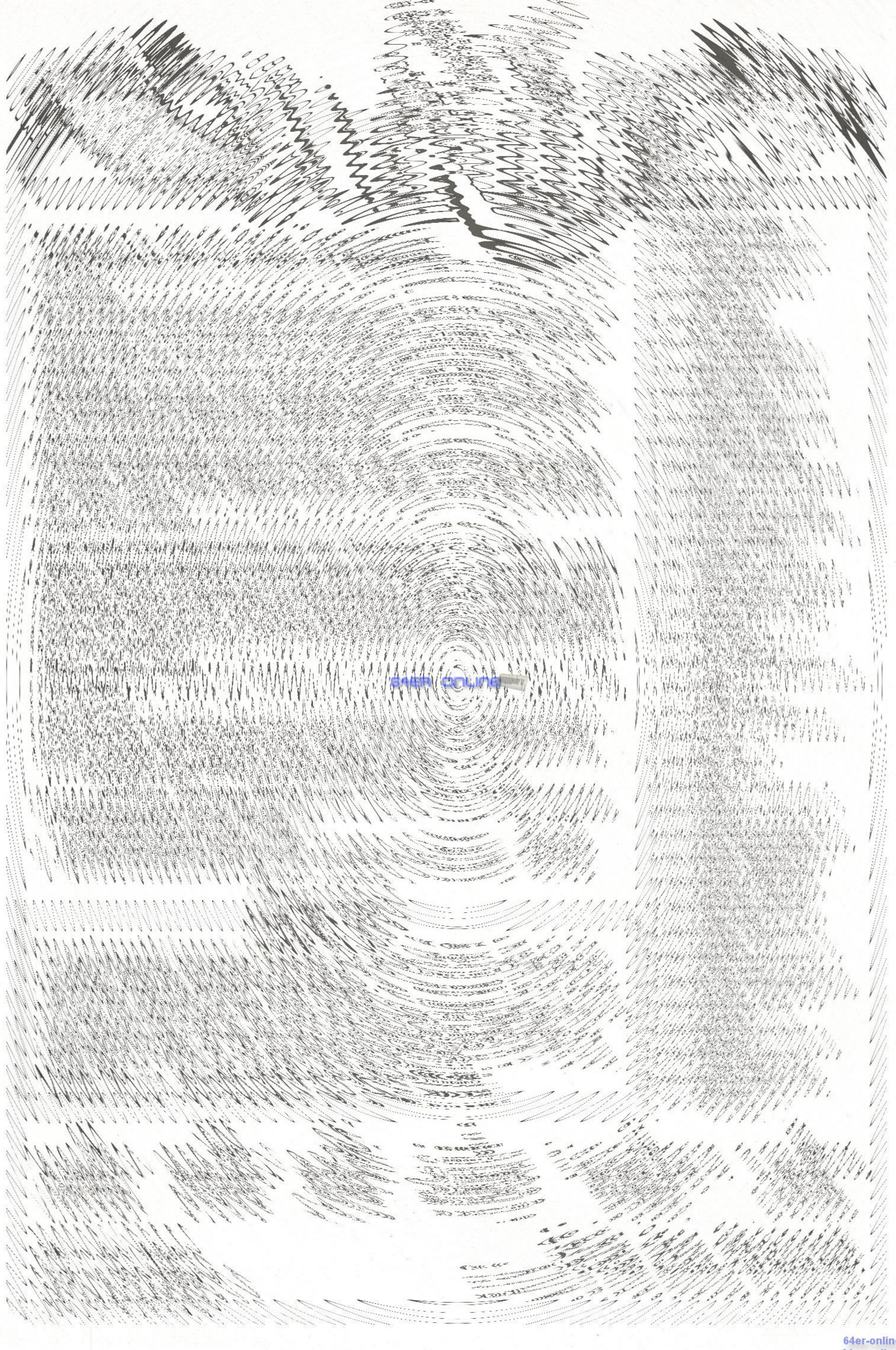
COMPUTER-MARKT

Wollen Sie einen gebrauchten Computer verkaufen oder erwerben? Suchen Sie Zubehör? Haben Sie Software anzubieten oder suchen Sie Programme oder Verbindungen? Der COMPUTER-MARKT von »64'er« bietet allen Computerfans die Gelegenheit, für nur 5,— DM eine private Kleinanzeige mit bis zu 5 Zeilen Text in der Rubrik Ihrer Wahl aufzugeben. Und so kommt Ihre private Kleinanzeige in den COMPUTER-MARKT der **Juni-Ausgabe** (erscheint am 16. Mai 86): Schicken Sie Ihren Anzeigentext bis zum 16. April 86 (Eingangsdatum beim Verlag) an »64'er«. Später eingehende Aufträge werden in der **Juli-Ausgabe** (erscheint am 13. Juni 86) veröffentlicht.

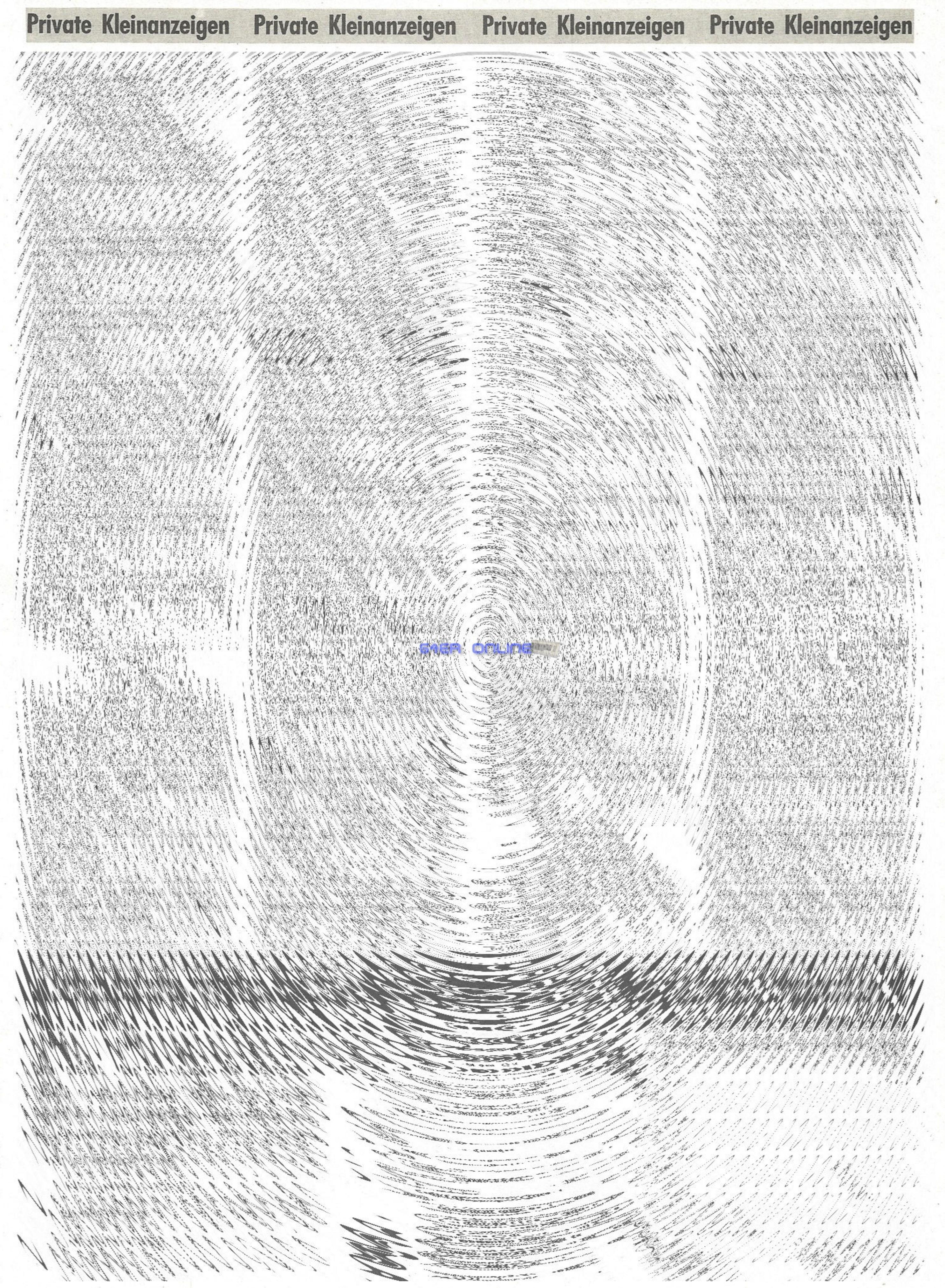
Am besten verwenden Sie dazu die vorbereitete Auftragskarte am Anfang des Heftes. Bitte beachten Sie: Ihr Anzeigentext darf maximal 5 Zeilen mit je 32 Buchstaben betragen. Überweisen Sie den Anzeigenpreis von DM 5,— auf das Postscheckkonto Nr. 14199-803 beim Postscheckamt mit dem Vermerk »Markt & Technik, 64'er« oder schicken Sie uns DM 5,— als Scheck oder in Bargeld. Der Verlag behält sich die Veröffentlichung längerer Texte vor. Kleinanzeigen, die entsprechend gekennzeichnet sind, oder deren Text auf eine gewerbliche Tätigkeit schließen läßt, werden in der Rubrik »Gewerbliche Kleinanzeigen« zum Preis von DM 12,— je Zeile Text veröffentlicht.

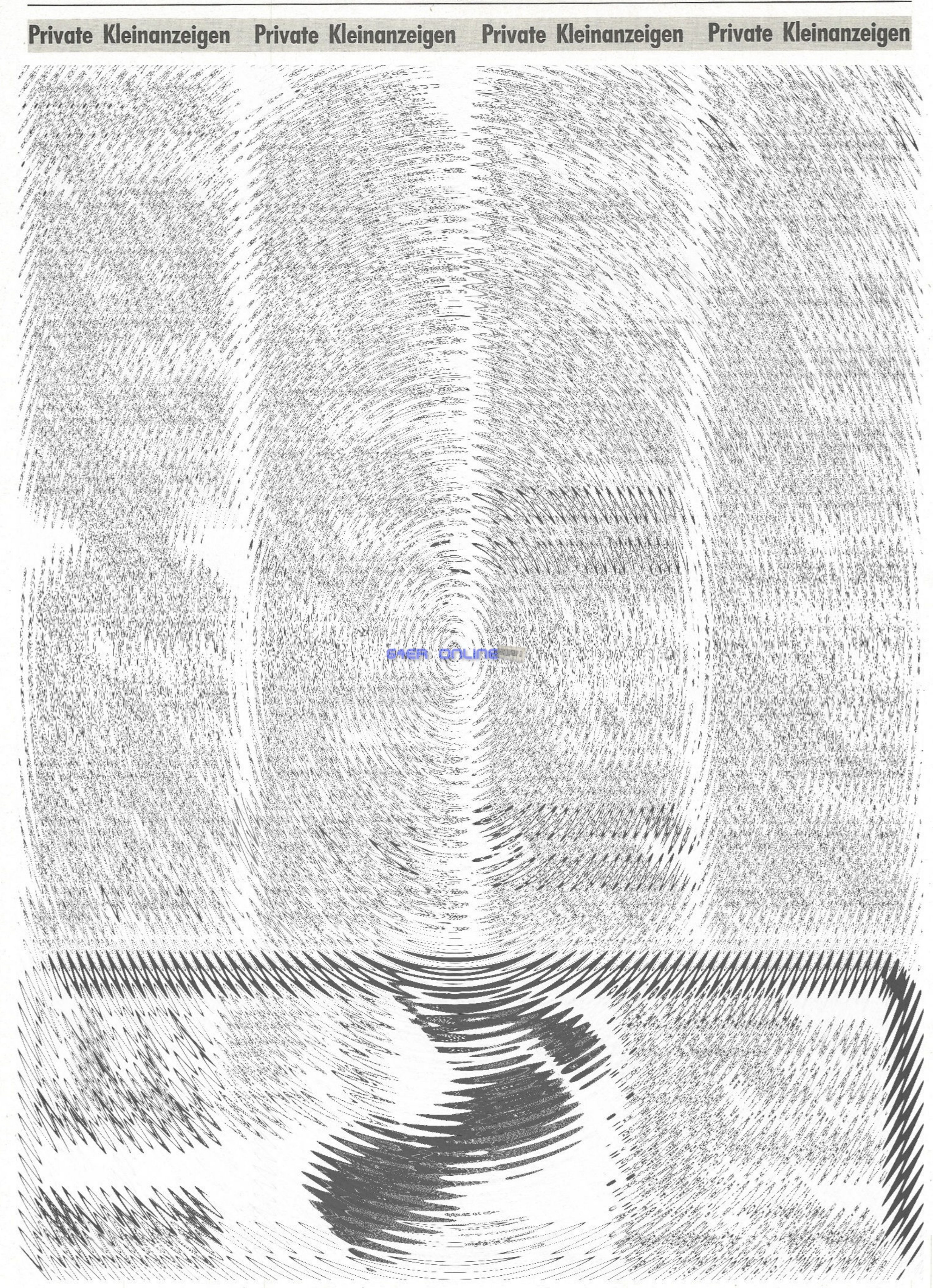
Private Kleinanzeigen

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen







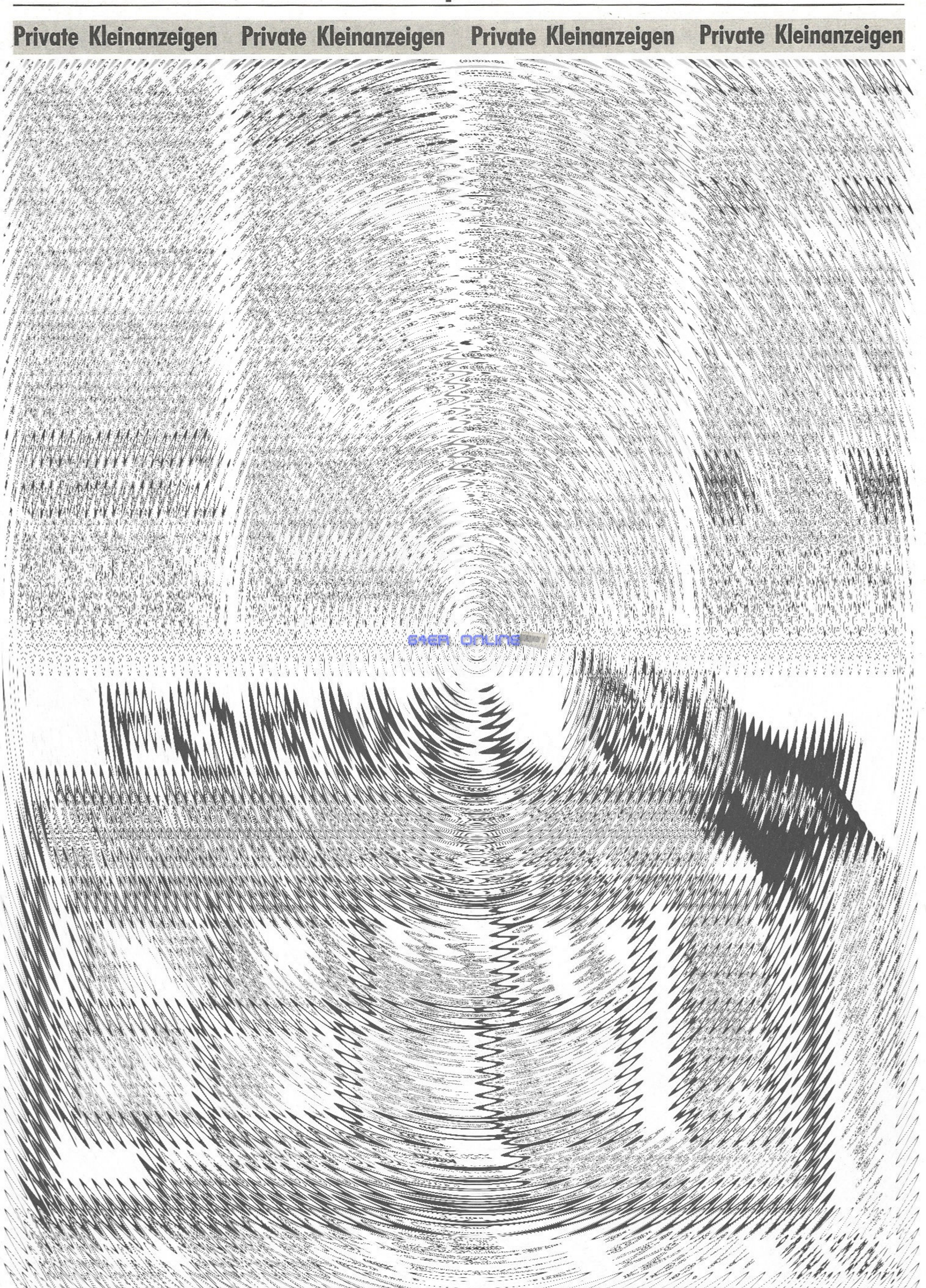




Private Kleinanzeigen Private Kleinanzeigen



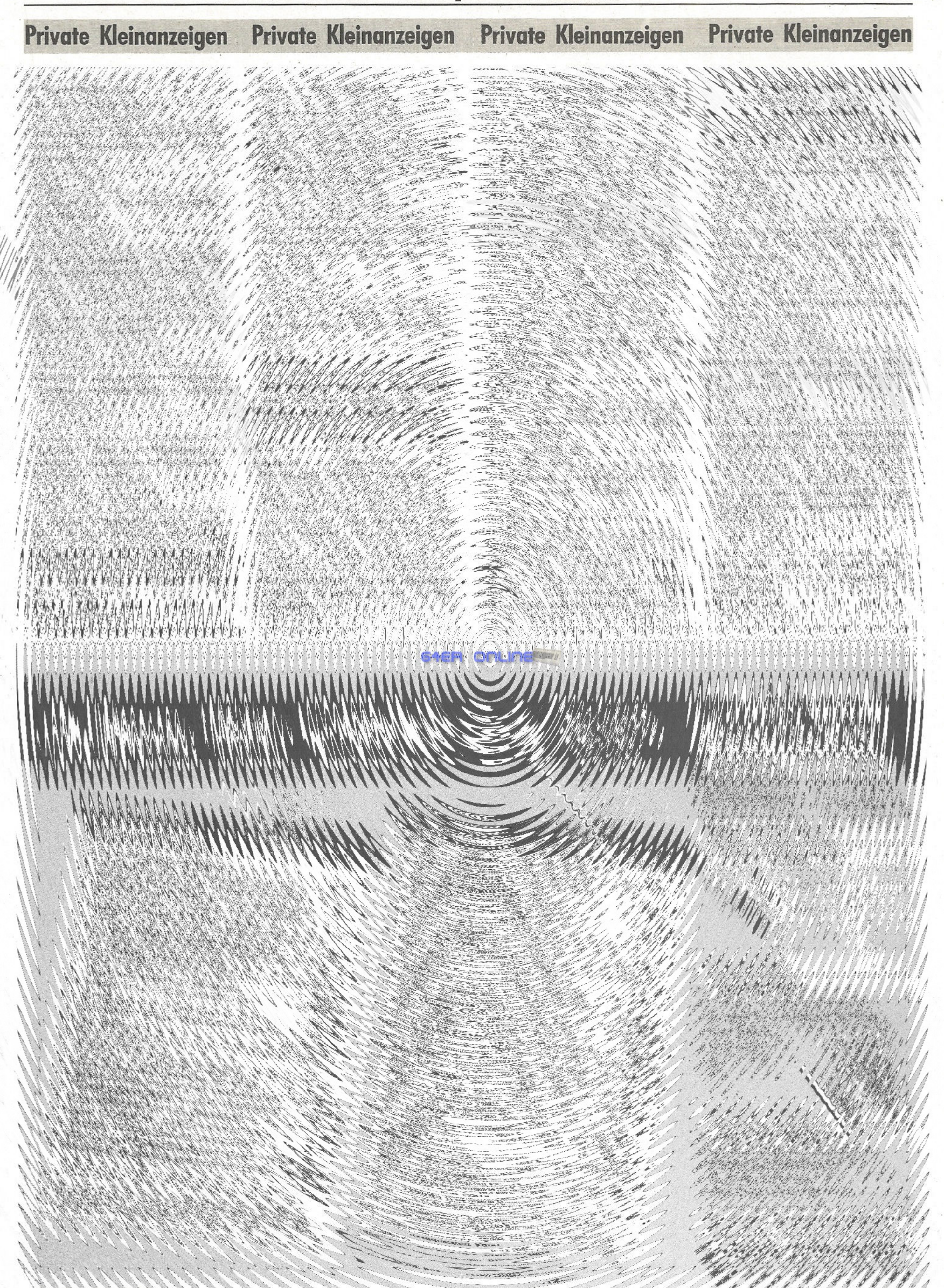
Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen



Private Kleinanzeigen Private Kleinanzeigen

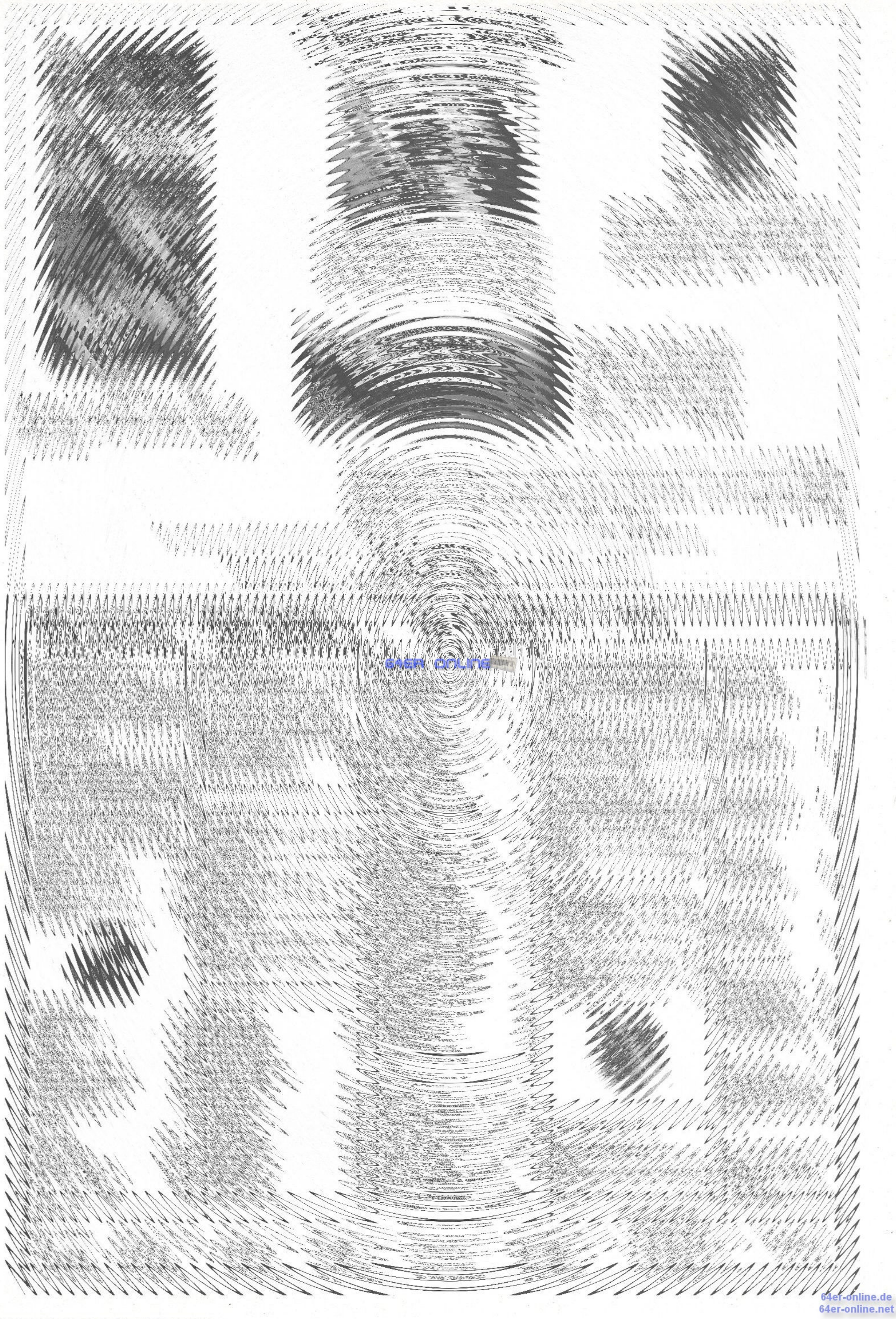


Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen



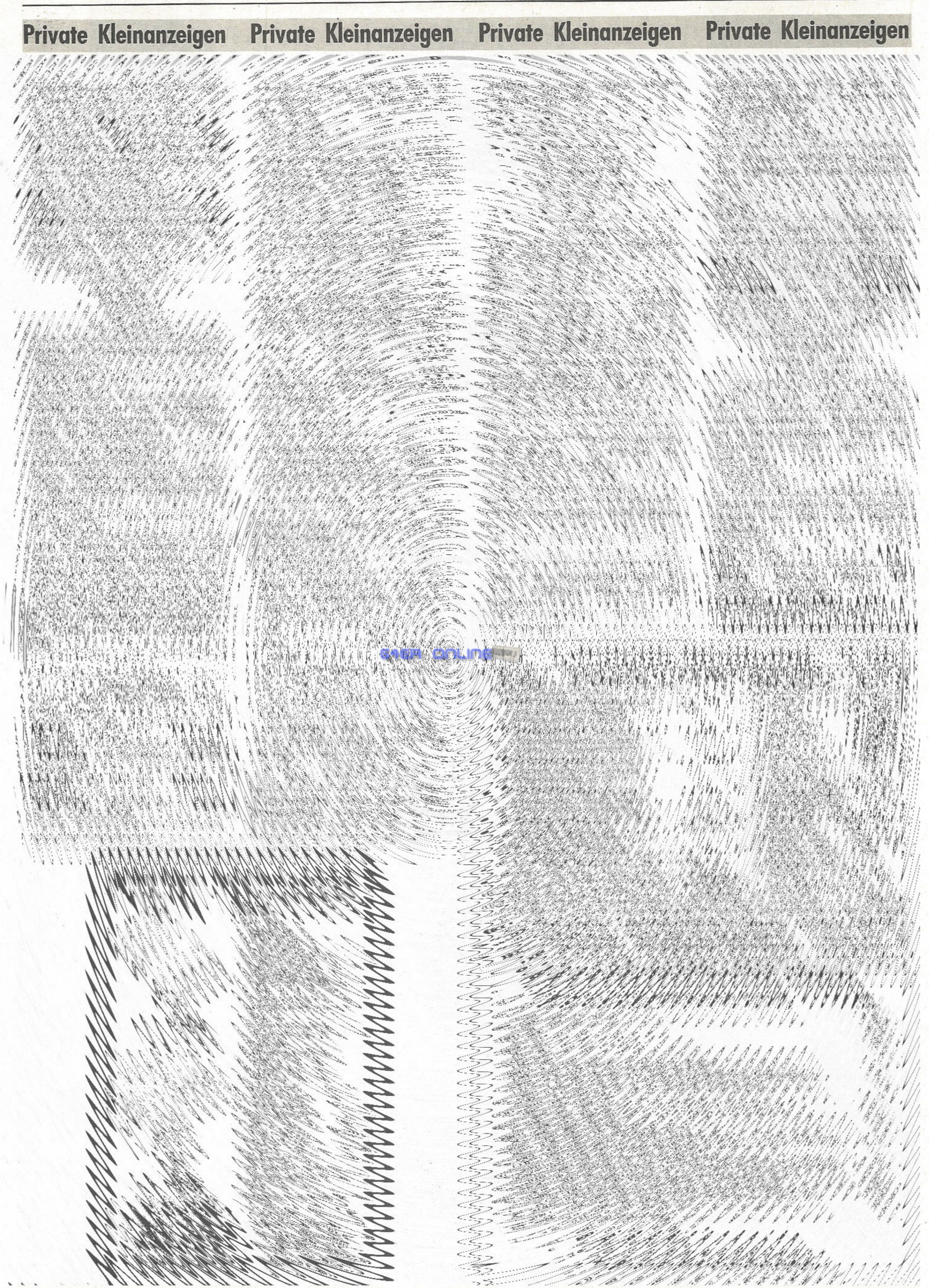
Private Kleinanzeigen Private Kleinanzeigen





Private Kleinanzeigen

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen



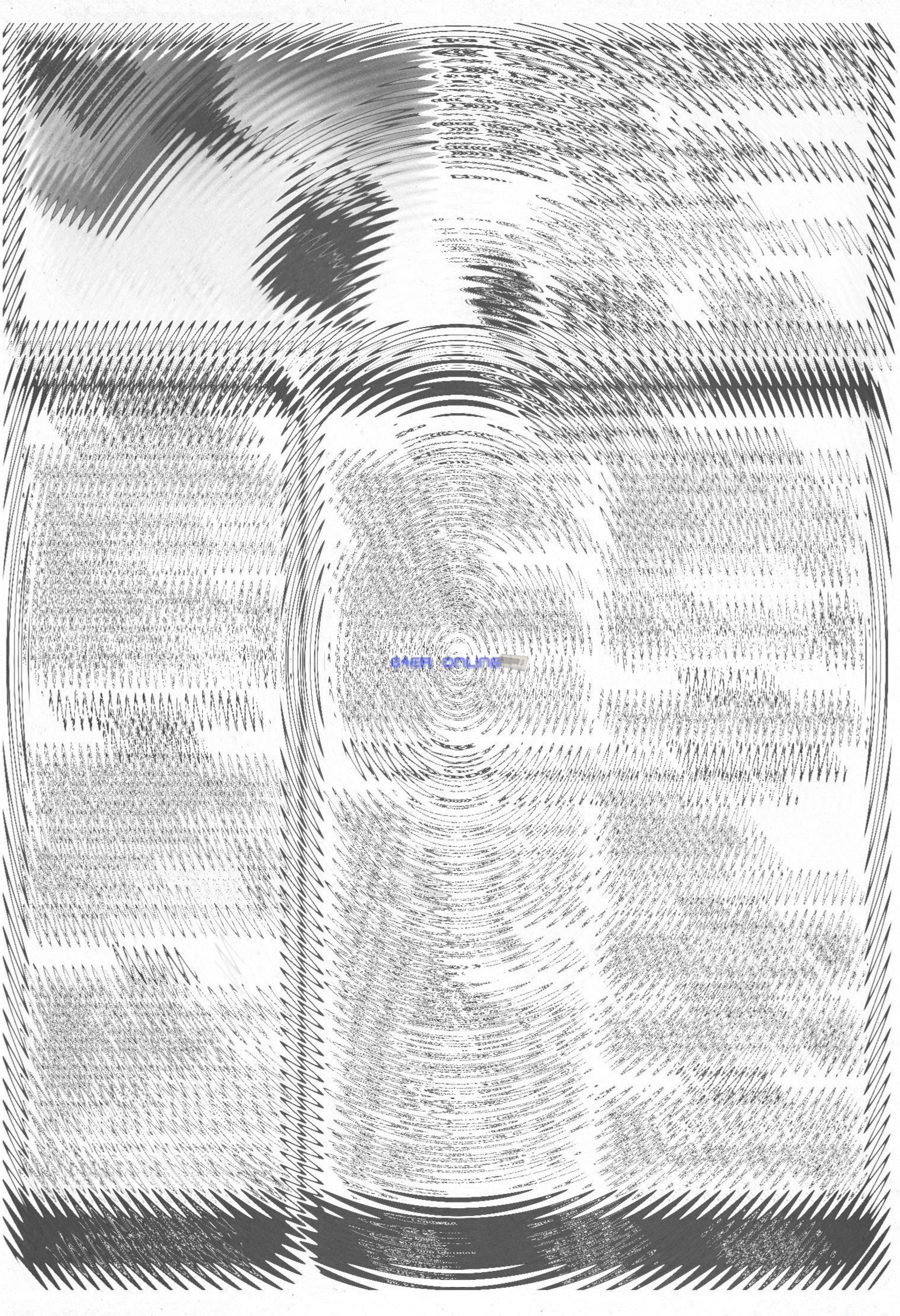
Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen





Private Kleinanzeigen Private Kleinanzeigen





Private Kleinanzeigen Private Kleinanzeigen

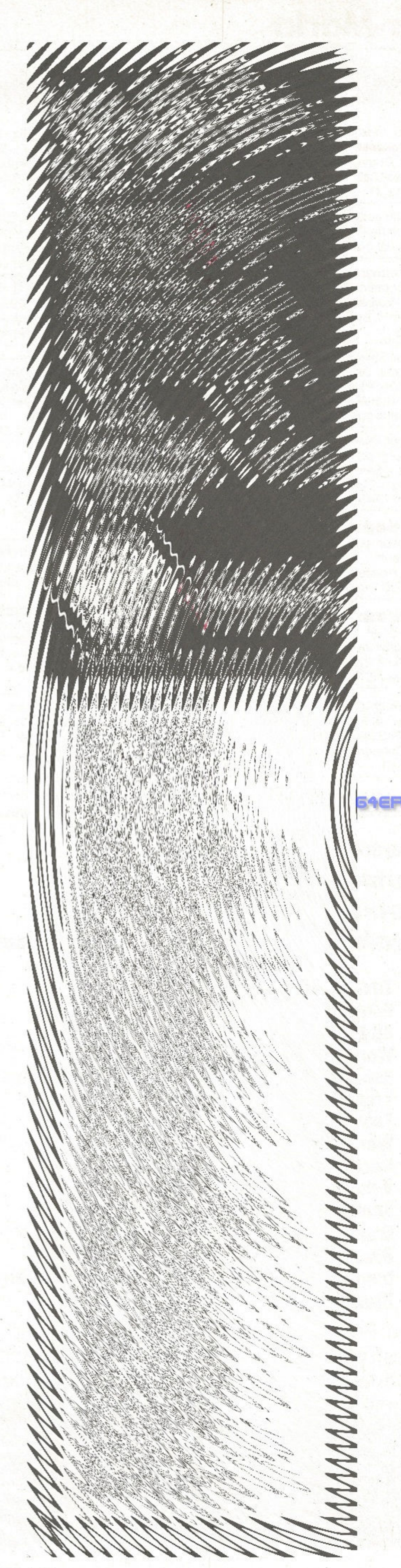
Private Kleinanzeigen

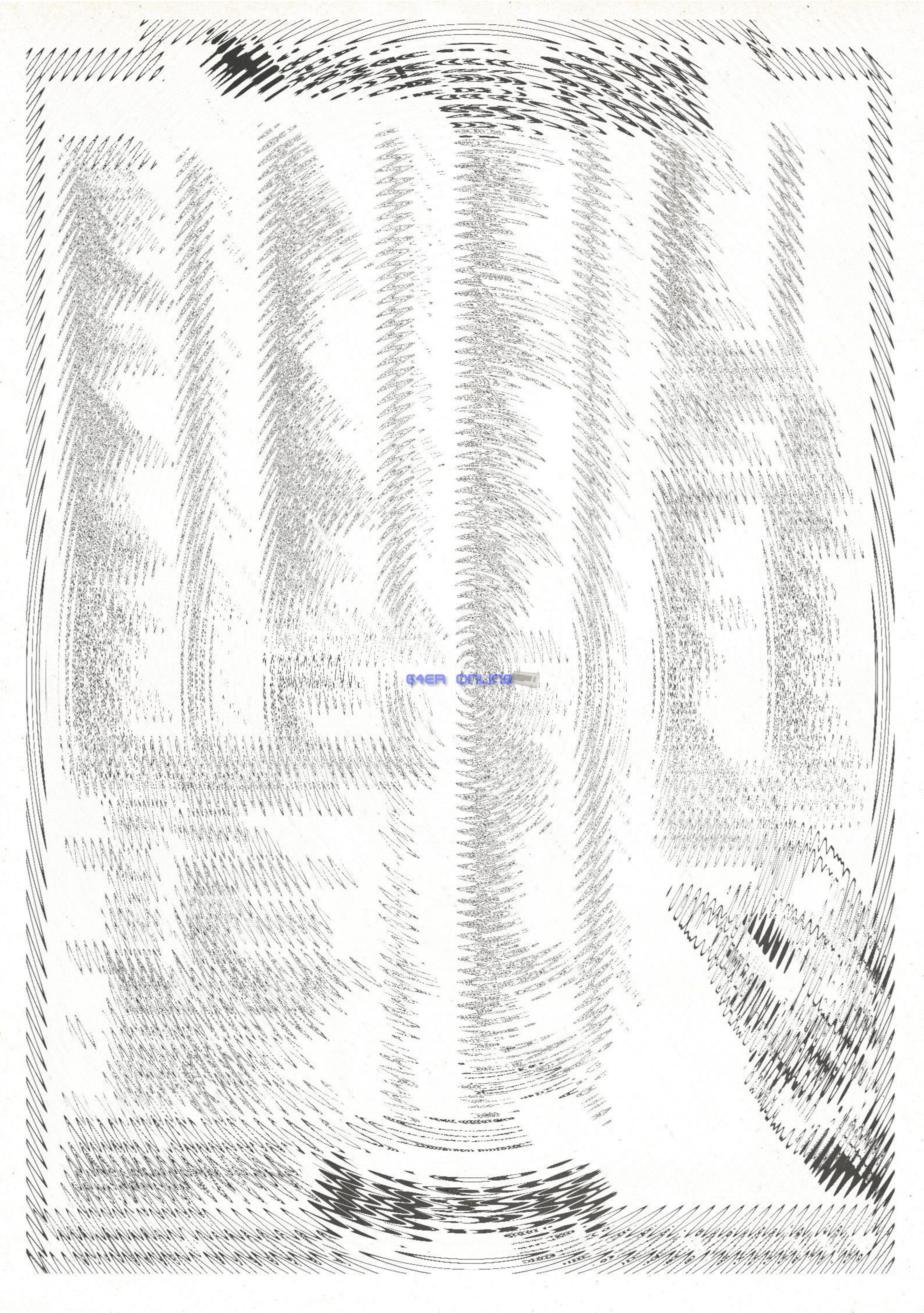
Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen



Private Kleinanzeigen

Private Kleinanzeigen



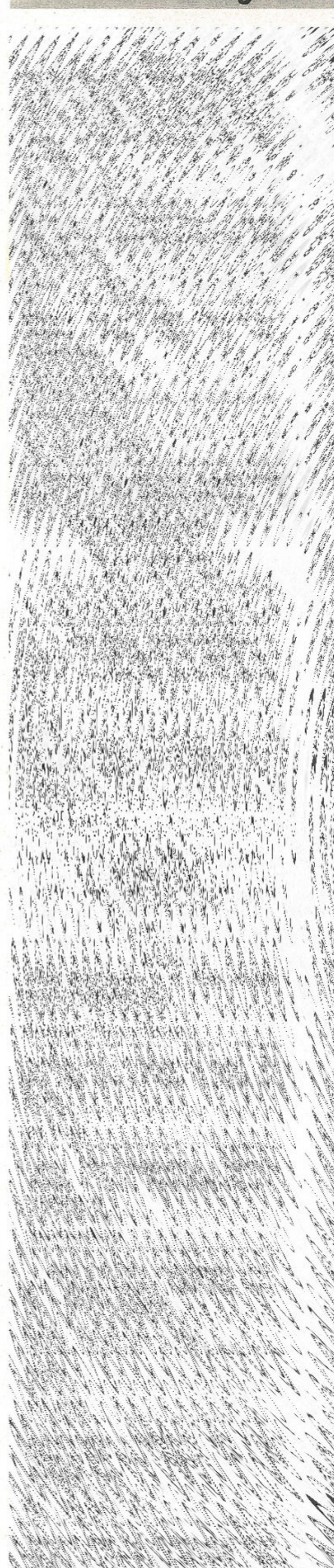


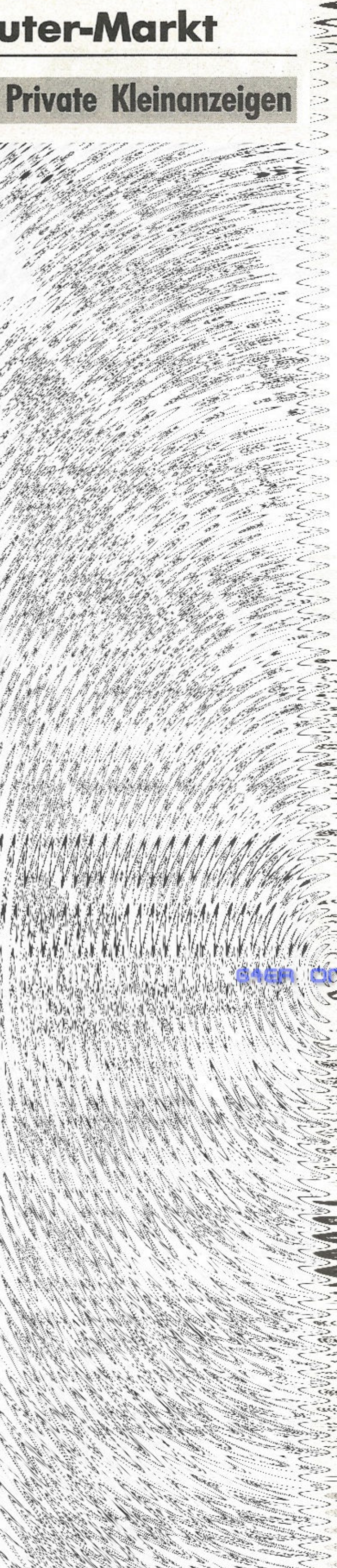


Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

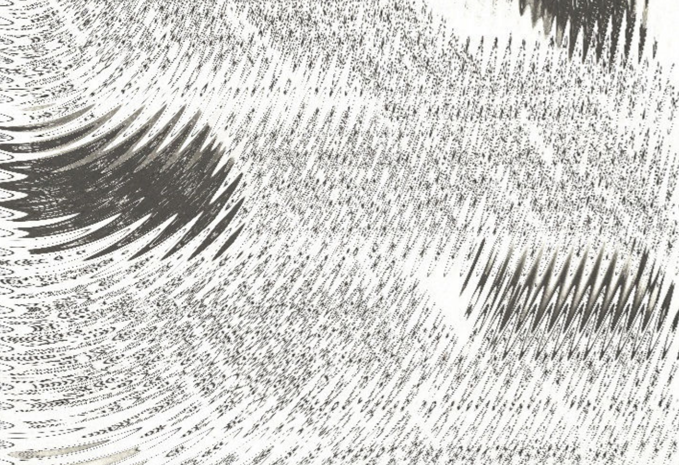



64ER ONLINE







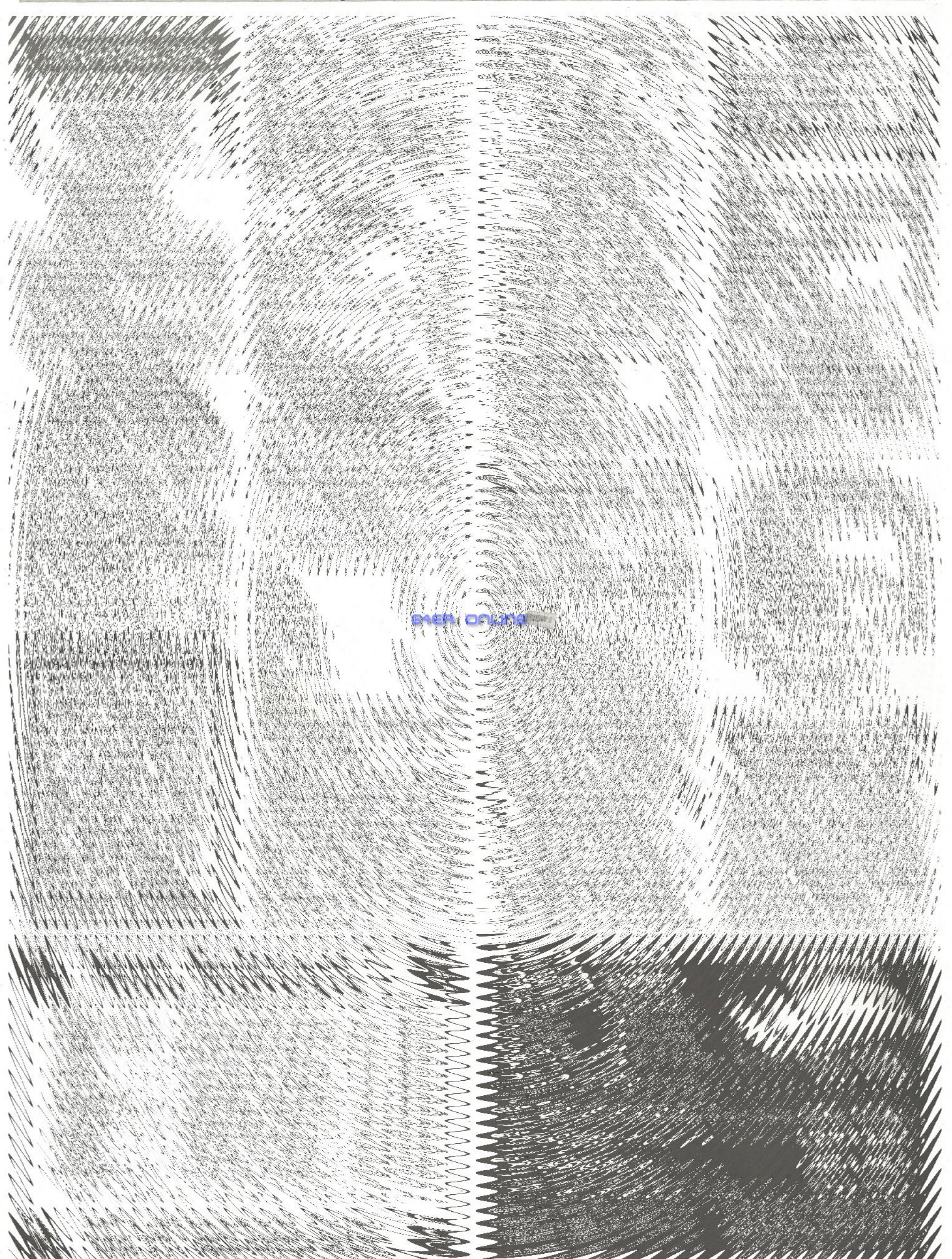





64ER ONLINE

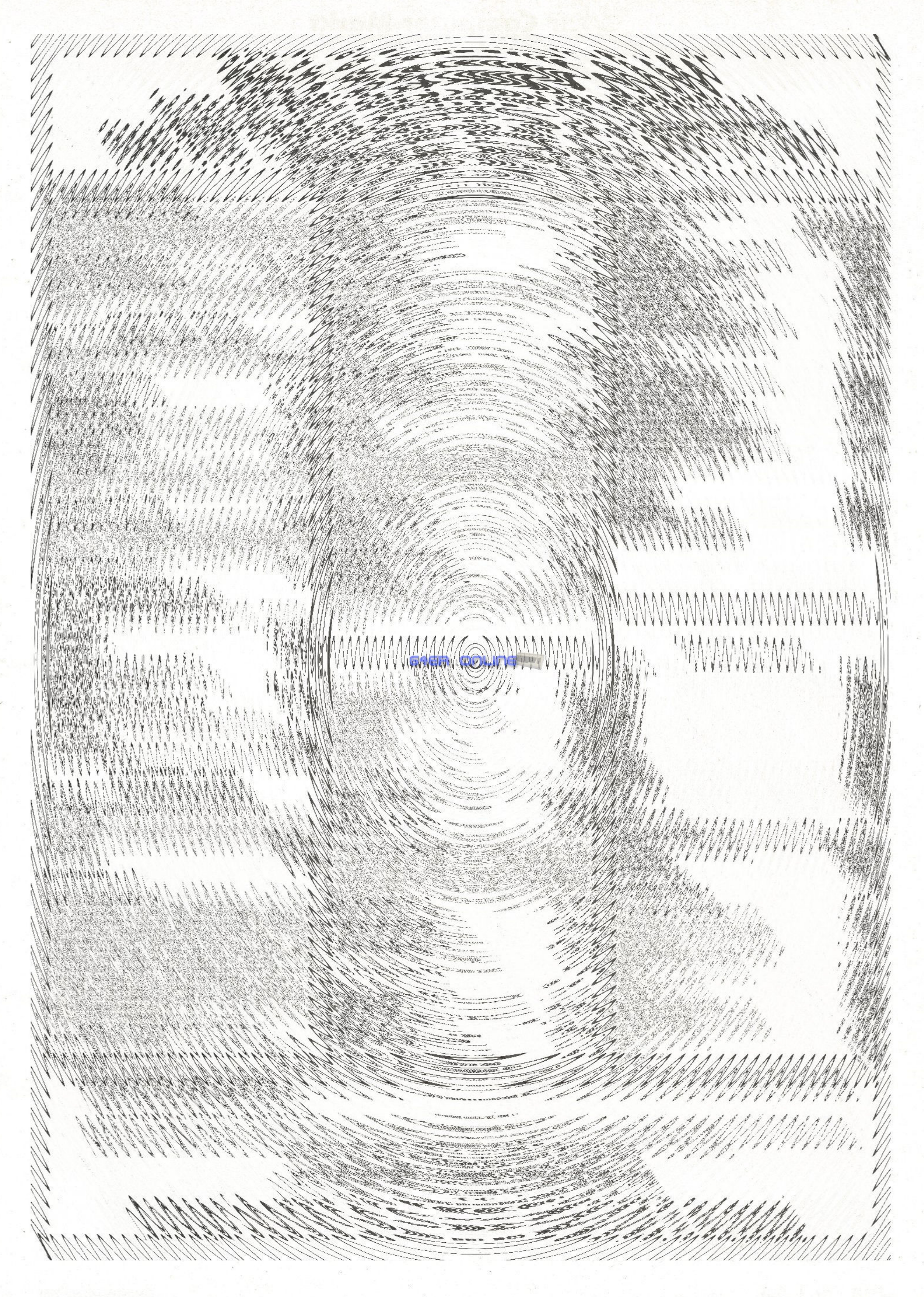
Gewerbliche Kleinanzeigen

Gewerbliche Kleinanzeigen



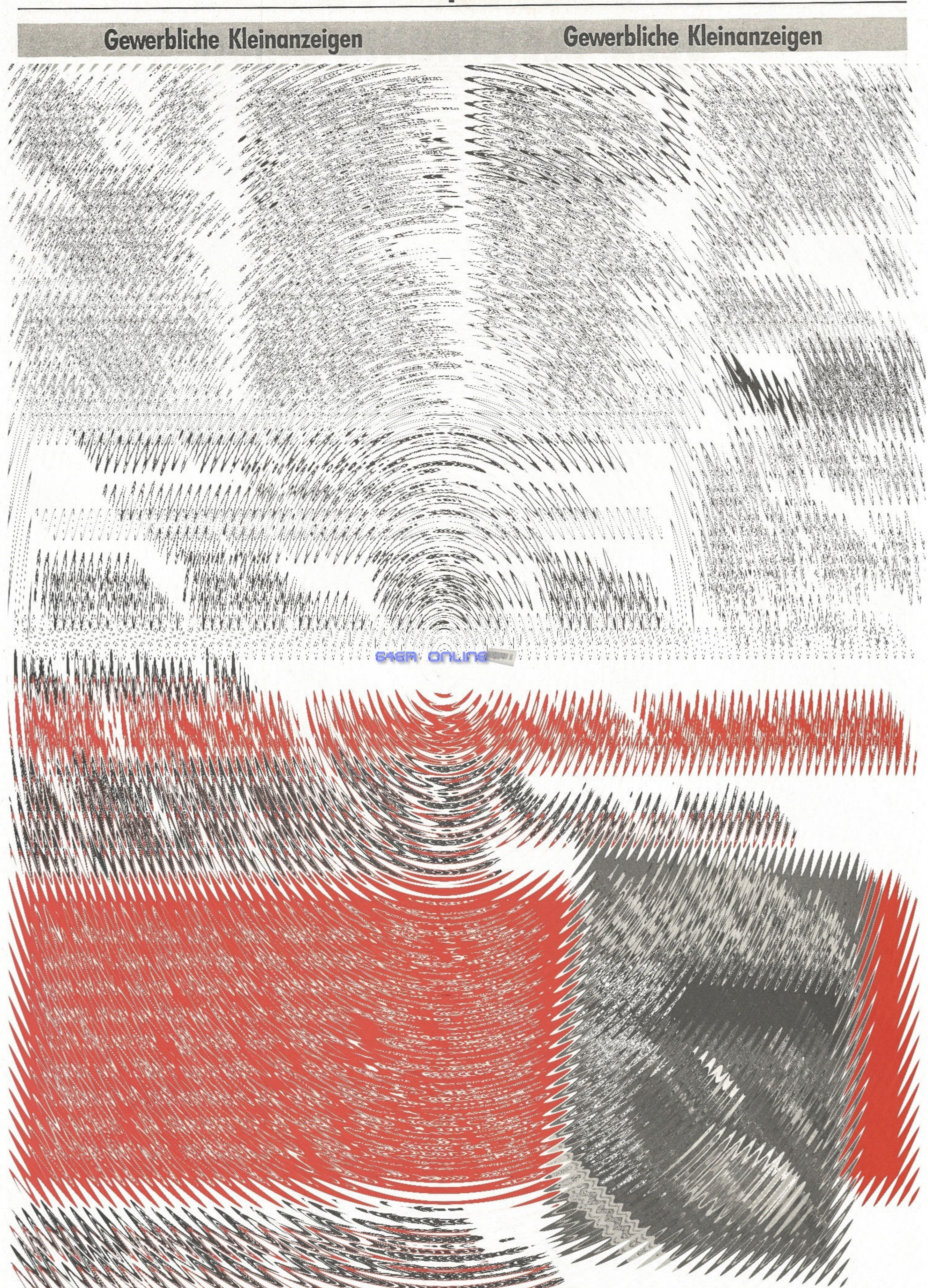
Gewerbliche Kleinanzeigen



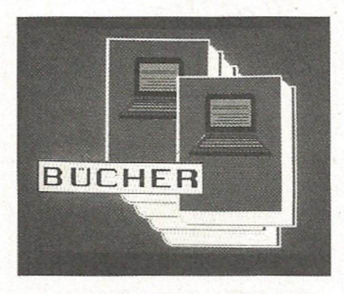


Gewerbliche Kleinanzeigen

Gewerbliche Kleinanzeigen







DER SICHERE EINSTIEG IN PASCAL

Dieses Buch beweist sehr überzeugend, daß nicht alle Theorie grau sein muß. Zudem hätte der Titel kaum besser gewählt werden können. Auf 188 Seiten liegt eine Schritt-für-Schritt-Einführung vor, die das Buch zu einem der wenigen echten Einsteigerwerke macht. Anhand praktischer Beispiele aus der Mathematik und der Datenverarbeitung werden zunächst die wichtigsten Sprachelemente erklärt. Positiv fällt dabei die Systematik auf, mit der die Spitzfindigkeiten der pascal'schen Syntax auseinandergenommen und später vertieft werden. Daß immerhin die ersten 80 Seiten ausschließlich linearen Programmen gewidmet sind, spricht für sich. Dabei kommt jedoch, wie man vielleicht vermuten möchte, keine Langeweile auf, denn die Beispiele sind anregend und die Schreibweise des Autors ist ebenso unterhaltend wie lehrreich. So werden zunächst die Grundbegriffe des Programmierens klargelegt und die Ein- und Ausgabebefehle sowie die Operationen mit den verschiedenen Datentypen auseinandergesetzt. Jeder Erklärung folgt eine abgewandelte Aufgabenstellung mit deren Lösung. Als weitere Sprachelemente werden Wiederholungs-, Schleifen- und Entscheidungsanweisungen besprochen. Ebenfalls erschöpfend behandelt werden Unterprogrammtechniken und die Behandlung ein- und mehrdimensionaler Datenfelder (Arrays). Der theoretische Teil des Buches schließt mit einem kurzen Einblick in die Matrizenrechnung ab. Dieses Kapitel soll die Anwendung von Arrays verdeutlichen. Allerdings fällt an dieser Stelle ein Programm mit elementaren Fehlern auf, die der Lernende in diesem fortgeschrittenen Stadium aber bereits ausfindig machen kann.

Ist der Leser dem Buch bis zu dieser Stelle intensiv gefolgt, verfügt er bereits über einen Basis-Sprachumfang, der es ihm ermöglicht, sich in jede Pascal-Version, auf jedem Computer einzuarbeiten und seine Kenntnisse zu vertiefen. Zwar bezieht sich das Buch und die Programme auf einem NCR-Kleincomputer, doch sind alle Beispiele allgemeingültig, da eben nur ein elementarer Grundwortschatz vermittelt wird.

Anhand von vier »Projekten« werden komplette Programme entwickelt. Hierbei wird reelle Entwicklungsarbeit von der Aufgabenstellung bis hin zur Eingabe des Programms geleistet, und ganz nebenbei werden die erworbenen Kenntnisse weiter vertieft. Daß bei einem derart umfassenden Lehrbuch ein Anhang zum schnellen Nachschlagen sowie ein Stichwortregister nicht fehlen, ist selbstverständlich.

Einziges Haar in der Suppe ist, daß auf die Entwicklung und Darstellung von Algorithmen ganz verzichtet wurde. Daß dieser Punkt gerade bei einer Sprache wie Pascal ausgelassen wurde, ist unverständlich.

Dem Gesamteindruck nach kann dieses Buch jedem Computer-Neuling ans Herz gelegt werden, insbesondere auch, da keine Parallelen zu anderen Sprachen aufgezeigt werden. Wer schon in einigen anderen Sprachen zu Hause ist, dem wird dieses Buch wegen der vielfach mangelnden Professionalität allerdings zu unergiebig sein.

(Matthias Rosin/ev)

Info: Rudolf Busch, Der sichere Einstieg in Pascal, Franzis Verlag, 188 Seiten, ISBN 3-7723-7861-7, Preis 38 Mark

MINI-CAD MIT HI-EDDI PLUS

Welcher 64'er-Leser kennt nicht das mit dem Titel »Listing des Monats« prämierte Zeichenprogramm Hi-Eddi? Dieser überragende Grafik-Editor fand auch schnell seine Fans, und so kam es, daß der Autor Hans Haberl sich daran machte, sein Programm zu erweitern und ein Buch über Mini-CAD mit dem Hi-Eddi plus zu schreiben. Dieses Buch ist im Markt & Technik Verlag erschienen. »Buch« ist eigentlich eine glatte Untertreibung. Normalerweise wird zu einer Software ein Anleitungsheft geliefert. Nicht so beim Hi-Eddi plus. Hier wird einem 230 Seiten starken Buch eine komplette Programm-Diskette mit dem Zeichenprogramm selbst, vielen nützlichen Utilities und allen beschriebenen Grafik-Files beigelegt, und zwar im Preis inbegriffen. Doch nun zum Inhalt.

Auf die Einleitung folgt zunächst eine ausführliche Bedienungsanleitung, in der alle Funktionen und Befehle des Hi-Eddi plus aufgeführt und beschrieben sind. Daran schließt sich ein Kapitel an, in dem eine Vielzahl sinnvoller Anwendungen des Hi-Eddi plus aufgeführt werden. Vom Erstellen von Schaltplänen über Struktogramme und mehrteilige Bilder bis hin zu Zierschrift und Glückwunschkarten ist alles möglich. Das Erstellen von Schaltplänen wird von sogenannten »Construction Sets« unterstützt, in denen alle Schaltsymbole für analoge und digitale Elektronik vorhanden sind. Wenn man nun eine Schaltung auf die Mattscheibe zeichnen will, müssen nur die nötigen Symbole ausgewählt und richtig plaziert werden.

Auch die Liebhaber von kunstvollen Zierschriften kommen nicht zu kurz, denn es befinden sich einige vorgefertigte Construction Sets für Schriften auf der Diskette. Auf diese Weise ist es sehr einfach, Schilder oder Glückwunschkarten herzustellen. Zur Abrundung des Gesamtbildes ist noch ein Construction Set mit mannigfaltigen Rahmenmustern vorhanden.

Auch Struktogramme und Programmablaufpläne sind mit dem Hi-Eddi problemlos realisierbar. Und das sind längst noch nicht alle Bereiche, die von diesem Allround-Programm abgedeckt werden. Hierzu sei nur noch bemerkt, daß das Ausdrucken von Grafiken, die mehr als einen Bildschirmausschnitt umfassen, ebenfalls unterstützt wird. Es existieren Druckroutinen für die Commodore-Drukker VC1525, MPS-801 und 803. Auch Epson-Drucker und Kompatible können angesteuert werden. Mit der Druckeranpassung beschäftigt sich ein eigenes Kapitel.

Nun wird's spannend, denn auf der Disk befinden sich noch ein paar Erweiterungen zum Hi-Eddi plus. Eine davon ist der Bitmap-Compander. Er hat die Aufgabe, die Grafikbilder zu komprimieren. Dadurch passen mehr Bilder auf eine Diskette, da der Umfang der normal 33 Blöcke langen Bilder reduziert wird. Mit dem Extended Disk System lasen sich auch Hi-Eddi-Bilder in das Grafikformat anderer Zeichenprogramme umwandeln. Sehr sinnvoll ist auch das Adventure Graphic System, welches das Produzieren von Grafik-Adventures erleichtert.

Den Abschluß des Buches bildet eine Programmdokumentation für den Profi und eine Befehlsübersicht.

Fazit: Für den Preis von nur 48 Mark einschließlich Diskette mit Hi-Eddi plus und allen Hilfsprogrammen erhält man ein Buch, das eigentlich keinen Wunsch im Hinblick auf Grafik-Herstellung mehr offenlassen sollte.

Besonders erfreulich für den ehrlichen Käufer und Anwender ist die Tatsache, daß Markt & Technik auch bei dieser sehr preiswerten Buch/Software-Kombination auf einen Kopierschutz für die Diskette verzichtete. So kann der Benutzer sich eigene Sicherungskopien herstellen, statt gegen saftige Aufpreise solche beim Hersteller anzufordern. Es wäre wünschenswert, wenn dieses benutzerfreundliche Beispiel Schule machen würde.

(Michael Marek/ev)

Info: Hans Haberl, Mini-CAD mit Hi-Eddi plus, Markt & Technik Verlag AG, 230 Seiten, ISBN 3-89090-136-0, Preis 48 Mark einschließlich Programm-Diskette

WORDSTAR FÜR DEN C 128

Wieder einmal steht man vor einem Programm und seiner zwar ausführlichen, aber zur schnellen Einarbeitung nicht geeigneten Anleitung. Man kennt vom Hörensagen die Möglichkeiten und kann sie mangels Information nicht ausnutzen. Dieses Buch schließt die Lücke und bietet einen schnellen, effizienten Einstieg in Wordstar. Damit auch der wirkliche Einsteiger nicht gleich nach den ersten Seiten resigniert aufgibt, beginnt das Buch wirklich noch einmal mit den Grundlagen, deren Verständnis für den Umgang mit einem Computer vonnöten ist.

Das nächste Kapitel gibt nun einen kurzen Überblick über die Grundfunktionen von Wordstar, die dann in den einzelnen Abschnitten des Buches mit vielen Beispielen sehr ausführlich erläutert werden. Ein besonderer Augenmerk wird auf den wichtigsten Teil einer Textverarbeitung gelegt, der Ausgabe auf dem Drucker. Hier finden sich fast alle nur denkbaren Einstellungen aufgeführt, die ein Drucker oder eine Schreibmaschine nur ausführen kann.

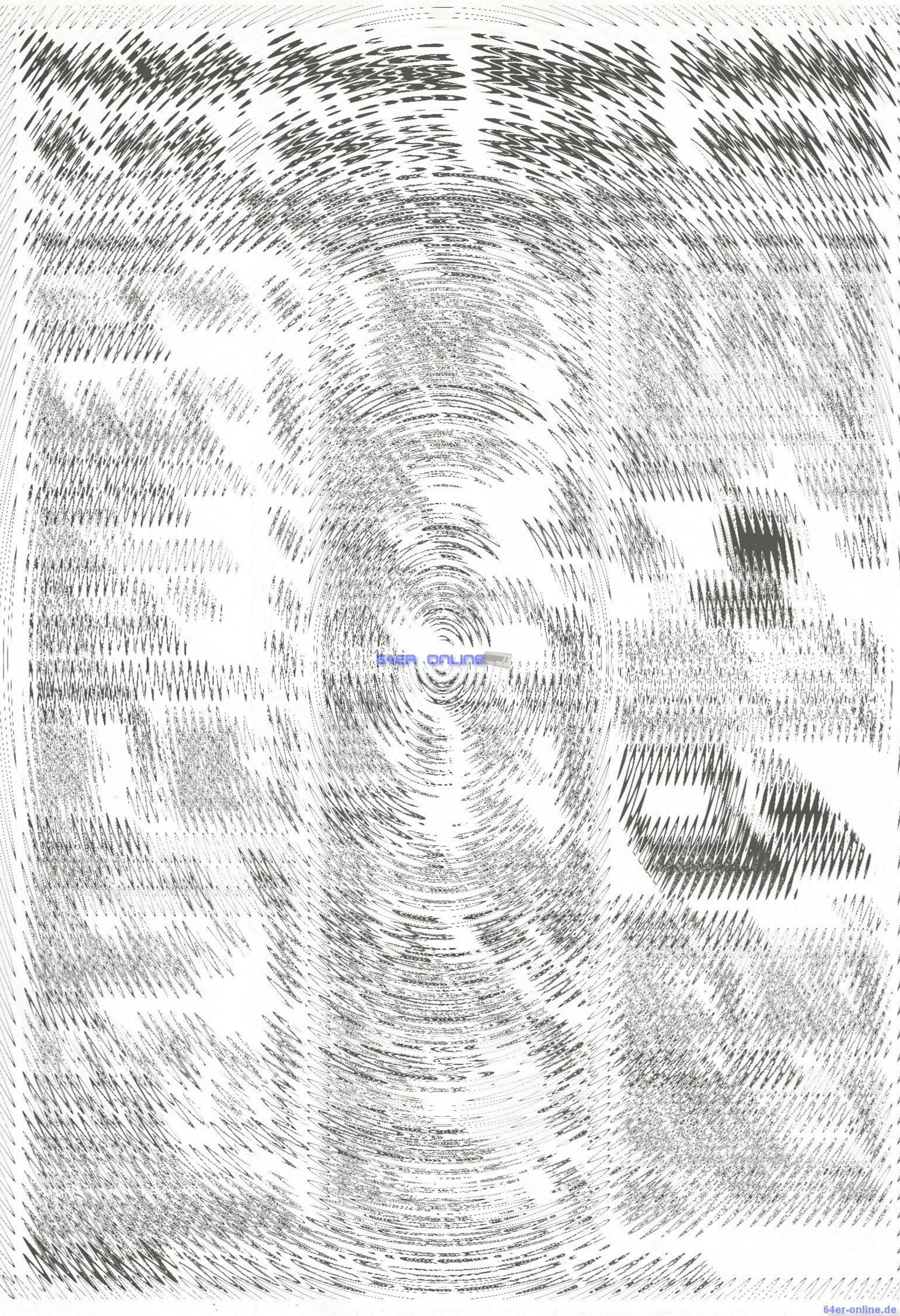
Der Umgang mit Mailmerge, einem Programm zur Erstellung von Serienbriefen, wird ebenso wie die Schnittstelle zu dBase II und Multiplan genau und leicht verständlich beschrieben.

Auch der Freak findet im letzten Kapitel genügend Material. So ist von der noch einfachen Druckeranpassung bis zur direkten Anpassung durch Patchen, das durch eine gut dokumentierte Adressenliste unterstützt wird, wirklich alles gegeben. Der Anhang ist als Gesamtübersicht ausgelegt, in ihm finden sich die Auflistung aller Menüs, Fehlermeldungen und Befehle. Weiterhin findet man eine Übersicht über die Tastaturbelegung des C 128, die gerade dem Anfänger eine große Hilfe ist. Fazit: Sucht man nach einem umfassenden Werk, das nach der Einarbeitungsphase nicht nutzlos im Bücherregal verschwindet, so ist man mit diesem Buch gut beraten.

(Udo Reetz/ev)

Info: Günter Jürgensmeier, Wordstar für den Commodore 128 PC, Markt & Technik Verlag AG, 436 Seiten, ISBN 3-89090-181-6, Preis: 49 Mark





Pascal-Kurs für Anfänger: Teil 2

Jetzt zeigt sich Pascal von der besten Seite: Programme können einfach, klar und problemnah formuliert werden. Pascal hat die Anweisungen dafür.

Dieser zweite Teil unseres Pascal-Kurses stellt die Anweisungen Goto und For, die If-Then-Else-Schleife, die Repeat- und die While-Schleife vor. Danach geht er auf die Verbundanweisung (Begin ... End) ein, mit deren Hilfe man einen Block aus Anweisungen zu einer einzigen Anweisung zusammenschließen kann.

Bevor ein Programm ablaufen kann, muß es in den Computer eingegeben werden. Beim Oxford- und beim Profi-Pascal-Compiler wird ein Editor verwendet, der dem Basic-Editor ziemlich gleicht. Oxford-Pascal verwendet weitgehend die Funktionen des Basic-Editors. Vor jeder Zeile des Pascal-Programms muß eine Zeilennummer stehen, die aber innerhalb von Pascal keinerlei Bedeutung hat. Profi-Pascal verwendet einen eigenen Editor, der ebenfalls zeilenorientiert arbeitet. Er ist in Pascal geschrieben und kann von einem fortgeschrittenen Programmierer geändert werden — der Quellcode wird mitgeliefert.

Nach der Eingabe sollte das Quellprogramms auf der Diskette abgespeichert werden. Bei Profi-Pascal ist dies unbedingt nötig, da der Compiler gesondert in den Arbeitsspeicher geladen wird. Der Compiler übersetzt das Programm und gibt den Objekt-Code auf die Diskette aus. Syntax-Fehler werden dem Benutzer auf dem Bildschirm mitgeteilt. Der Compiler wartet dann darauf, daß der Programmierer die Leertaste drückt, damit die Übersetzung fortgesetzt werden kann. Die Stop-Taste führt dazu, daß in den Editor verzweigt wird. Nach einer fehlerfreien Übersetzung kann das Programm gestartet und ausgeführt werden.

Bei Oxford-Pascal wird zwischen einem Resident-Modus und einem Disk-Modus unterschieden. Der Vorgang beim Disk-Modus gleicht dem Verfahren bei Profi-Pascal. Beim Resident-Modus kann direkt übersetzt und ausgeführt werden, ohne daß auf eine Diskette zugegriffen wird. Ein Abspeichern des Programms ist nicht notwendig, aber empfehlenswert. Oxford-Pascal weist eine größere Ähnlichkeit mit Commodore-Basic auf als Profi-Pascal, weil es das gleiche Betriebssystem benutzt.

Die Anweisungen von Pascal kann man in einfache und in strukturierte Anweisungen einteilen (siehe Bild 1). Von strukturierten Anweisungen spricht man deswegen, weil diese Anweisungen wieder weitere Anweisungen enthalten dürfen.

Einfache Anweisungen

Von den einfachen Anweisungen wurde im ersten Teil bereits die Wertzuweisung besprochen. Sie besitzt kein eigenes Schlüsselwort. Wertzuweisungen besitzen folgende allgemeine Form:

```
Variable := Ausdruck
```

Dabei gilt, daß Variable und Ausdruck in der Regel vom gleichen Datentyp sein müssen. Eine Ausnahme bildet lediglich die Zuweisung eines Integer-Ausdrucks auf eine Real-Variable. Weil hier kein Informationsverlust auftreten kann, ist diese Zuweisung erlaubt. Die Bildung von komplexeren Ausdrücken wird im nächsten Teil behandelt.

Eine Anweisung, die in Basic sehr wichtig ist, sei hier nur am Rande erwähnt: die Goto-Anweisung. Während man in Basic kaum ohne sie auskommen kann, wird sie in Pascal nicht benötigt. Manche Umsteiger haben aber Probleme, ihren Basic-Stil zu vergessen und »Pascal-like« zu programmieren. Sie sollten bei der Verwendung der Goto-Anweisung das folgende beachten:

Auch wenn innerhalb der beiden Editoren eine Zeilennummer einzugeben ist, wird diese Zeilennummer nicht als Sprungmarke verwendet. Der Anwender muß alle Sprungziele, die in Pascal als Label bezeichnet werden, selbst definieren. Als Label sind ganze Zahlen ohne Vorzeichen erlaubt. Bei Profi-Pascal sind dies die Zahlen 0 bis 32767, bei Oxford-Pascal darf die Zahl sogar aus acht Ziffern bestehen. Die Labeldeklaration muß sofort nach dem Programmnamen stehen. Beispiel:

```
program springe;
  label 5,10;
  var i:integer;
begin
  i:=1;
  goto 5;
  10: writeln('Marke 10');
  5: i:=i+1;
  if i<=10 then goto 10
end.
```

Die Verbund-Anweisung

In einem Pascal-Programm stehen alle Anweisungen zwischen (mindestens je) einem Begin und einem End. Zusammen bilden die von Begin und End eingeschlossenen Anweisungen den Anweisungsteil eines Programms, der dem Definitionsteil folgt. Die Block-Klammern Begin und End bezeichnet man auch als Verbundanweisung, weil alle Anweisungen zwischen diesen beiden »Marken« als eine Einheit (ein Block) angesehen werden. Sie werden in Pascal wie ein Befehl behandelt. Anweisungen zwischen Begin und End werden in der Reihenfolge behandelt, in der sie aufgeführt sind. Daraus folgt, daß jedes Pascal-Programm eigentlich nur aus einer Verbund-Anweisung besteht.

Die Verbundanweisung wird benutzt, um Teile des Programms als eine Einheit zu erklären. Wo eine Anweisung stehen kann, darf statt dessen auch eine Verbundanweisung stehen. Dadurch wird erreicht, daß beispielsweise hinter einer If-Anweisung eine ganze Reihe weiterer Anweisungen stehen kann, ohne daß sie mit einem Goto übersprungen werden müssen:

```
 if a=x then
  begin
   a:=10;
   x:=5;
   writeln('irgendetwas')
  end;
a:=a+1; ....
```

In diesem Beispiel werden alle Anweisungen zwischen Begin und End nur dann ausgeführt, wenn die Bedingung »a = x« wahr ist. Sonst wird gleich mit der Anweisung nach End fortgefahren. Die Verbundanweisung ist also ein ganz wesentliches Hilfsmittel, um in Pascal strukturierte Programme (ohne Goto-Anweisung!) zu schreiben.

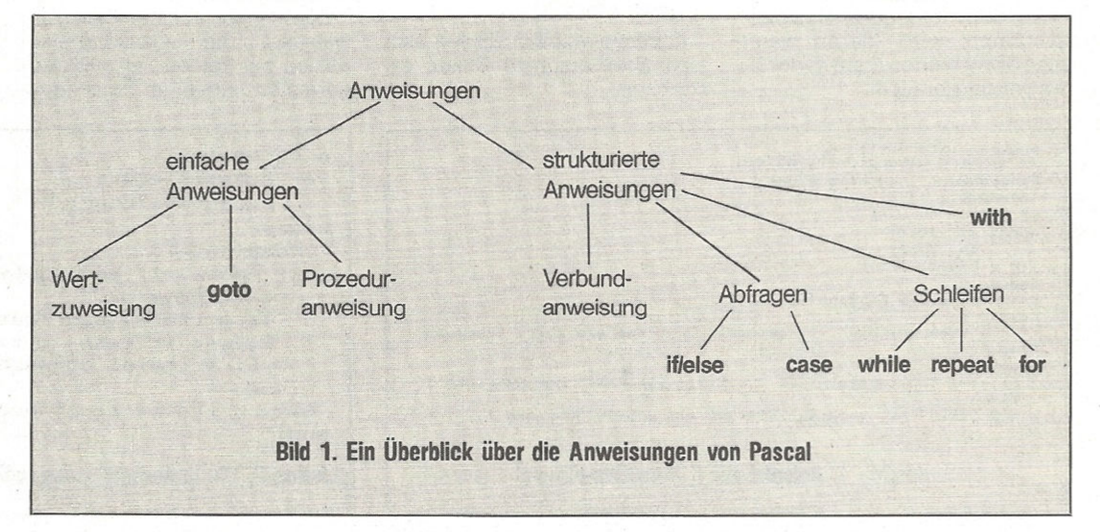


Bild 1. Ein Überblick über die Anweisungen von Pascal

In der Regel wird jede Anweisung mit einem Strichpunkt »;« abgeschlossen. Nur eine Ausnahme gibt es von dieser Regel: endet eine Anweisung direkt vor einem End, so kann der Strichpunkt weggelassen werden.

Die If-Anweisung sorgt für eine bedingte Programmausführung

Die If-Anweisung bewirkt, daß eine Anweisung nur dann ausgeführt wird, wenn eine bestimmte logische Bedingung erfüllt ist. Wenn die Bedingung nicht erfüllt ist, dann wird der Else-Teil interessant. Existiert kein Else-Teil (Fall 1), so wird keine weitere Anweisung im If-Then-Else-Block ausgeführt. Sonst ist die Anweisung, die im Else-Teil steht (Fall 2), die nächste auszuführende Anweisung. Es gibt also folgende zwei Formen der If-Anweisung:

Fall 1:

```
IF bed THEN aktion
```

Fall 2:

```
IF bed THEN aktion1 ELSE
aktion2
```

Im ersten Fall wird keine Anweisung ausgeführt, wenn der logische Ausdruck den Wert »false« ergibt. Im zweiten Fall wird dann die dem else folgende »aktion 2« ausgeführt. Als logischer Ausdruck wird ein Ausdruck bezeichnet, der als Ergebnis den Wert »false« oder »true« hat. Dies ist ein Vergleich oder ein Ausdruck mit Variablen vom Typ Boolean. Wichtig ist noch, daß unmittelbar vor einem else auf keinen Fall ein Strichpunkt stehen darf; dies würde zu einer Fehlermeldung führen. Die Syntax betrachtet nämlich die gesamte If/Else-Struktur als eine Anweisung und verbietet daher den Strichpunkt.

Bei der If/Else-Anweisung wird auf jeden Fall eine Anweisung ausgeführt, entweder »aktion 1« oder »aktion 2«.

An dem folgenden Beispielprogramm wird klarer, wann man diese beiden Formen der If-Anweisung braucht:

```
if a>=0 then a:=a+1
  else a:=0;
if (a<0) and (b>c) then
 begin
  a:=a+1;
  c:=b*a;
 end
  else
    begin
     writeln('a = ',a);
     writeln('b = ',b)
    end;
```

Das folgende Beispiel ist falsch!

```
var i:integer;
begin
 i:=10;
 if i then writeln;
end;
```

In diesem Fall ist die Bedingung kein logischer Ausdruck, sondern sie ist vom Typ Integer! Pascal unterscheidet im Gegensatz zu Basic genau zwischen den einzelnen Datentypen und läßt daher die If-Abfrage im letzten Beispiel nicht zu.

In Listing 1 wird nochmals an einem vollständigen Beispiel die Wirkung der If-Anweisung gezeigt. Das Programm Intervall überprüft, ob ein bestimmter Eingabewert sich innerhalb eines Intervalls befindet, oder ob er kleiner oder größer ist. Die Grenzen des Intervalls (a und b) werden als Konstante definiert. Die Konstantendeklarationen werden in Pascal nach der Label-Anweisung, aber vor der Typen- beziehungsweise Variablendeklaration angegeben. Konstanten bekommen wie Variablen einen Namen, ihr Wert läßt sich aber nicht mehr verändern. Das Programm zeigt gleichzeitig, wie in Pascal Eingabewerte überprüft werden.

Das Beispielprogramm enthält einen Kommentar. In Pascal werden Kommentare zwischen die Zeichen »(*« und »*)« eingeschlossen. Beispiel:

```
(* dies ist ein Kommentar *)
(* dies
  ist
  auch einer *)
```

Kommentarzeilen dürfen sich also über mehrere Zeilen erstrecken.

Die Case-Anweisung verarbeitet mehrere Alternativen

In manchen Fällen gibt es nicht nur zwei Alternativen, sondern gleich mehrere. In Basic verwendet man dann entsprechend viele, in sich verschachtelte If-Schleifen. Diese Vorgehensweise ist nicht nur schlecht lesbar, sondern auch fehleranfällig. Ein Pascal-Programm verwendet in einem solchen Fall die Case-Anweisung. In ihrer allgemeinen Form sieht sie folgendermaßen aus:

```
CASE ausdruck OF
 Case-Label1: Anweisung1;
 Case-Label2: Anweisung2;
 ....
 Case-LabelN: AnweisungN
 end;
```

Der Ausdruck nach »case« muß ein skalarer Type sein. Skalare Typen sind Integer, Boolean, Char und Aufzählungstyp. Der Aufzählungstyp wird später besprochen. Real ist nicht erlaubt. Ausdruck (zahl) und Case-Label (1, 2 und 3) müssen vom gleichen Typ sein.

Beispiel:

```
CASE zahl OF
 1: eins;
 2: zwei;
 3: BEGIN eins; zwei; drei
 END;
```

Vor einer Anweisung können mehrere Labels stehen, die durch Komma getrennt werden. Ein Beispiel dazu finden Sie im Programm »Fall« (Listing 2) beispielsweise in der Programmzeile

```
2,3,4: writeln (o'.k'.)
```

Ausgewählt und ausgeführt wird diejenige Anweisung, deren Case-Label mit dem Ergebnis des Ausdruck übereinstimmt. Listing 2 enthält ein Beispiel zur Case-Anweisung. Insgesamt werden drei Fälle unterschieden. Als Ausdruck wird hier ganz einfach eine Integer-Variable verwendet.

Die Marken in einer Case-Anweisung, die »Case-Labels«, dürfen auf keinen Fall mit den normalen Labels im Programm verwechselt werden. Sie brauchen nicht gesondert in einer Label-Deklaration definiert werden. Und ein Versuch, mit dem »goto« ein solches Case-Label anzuspringen, ist natürlich erfolglos!

Eine Frage ist noch offen: was passiert, wenn der Ausdruck einen Wert ergibt, für den kein Case-Label vorhanden ist? Dieser Fall ist in Standard-Pascal nicht definiert und wird daher von jedem Compiler anders behandelt. Bei Oxford-Pascal erhält man den Laufzeitfehler »CASE ERROR«. Bei Profi-Pascal wird das Programm mit der nächsten Anweisung ohne Fehlermeldung fortgesetzt.

Profi-Pascal besitzt eine weitere Variante der Case-Anweisung, mit der dieses Problem behandelt werden kann. Alle Fälle, die nicht in der Liste vorkommen, werden mit einem Else-Label abgefangen. Das sieht dann beispielsweise so aus:

```
case zahl of
 1: eins;
 2: zwei;
 3: begin eins; zwei; drei
 end
 else vier
 end;
```

In den Fällen, in denen der skalare Typ »zahl« ungleich 1,2 oder 3 ist, wird der Else-Zweig aufgerufen. Die Anweisungen hinter den Case-Labels in unserem Beispiel (nämlich »eins«, »zwei« und »drei«) sind Aufrufe von Unterprogrammen. In Pascal werden Prozeduren mit ihrem Namen aufgerufen.

Wiederholung mit der For-Anweisung

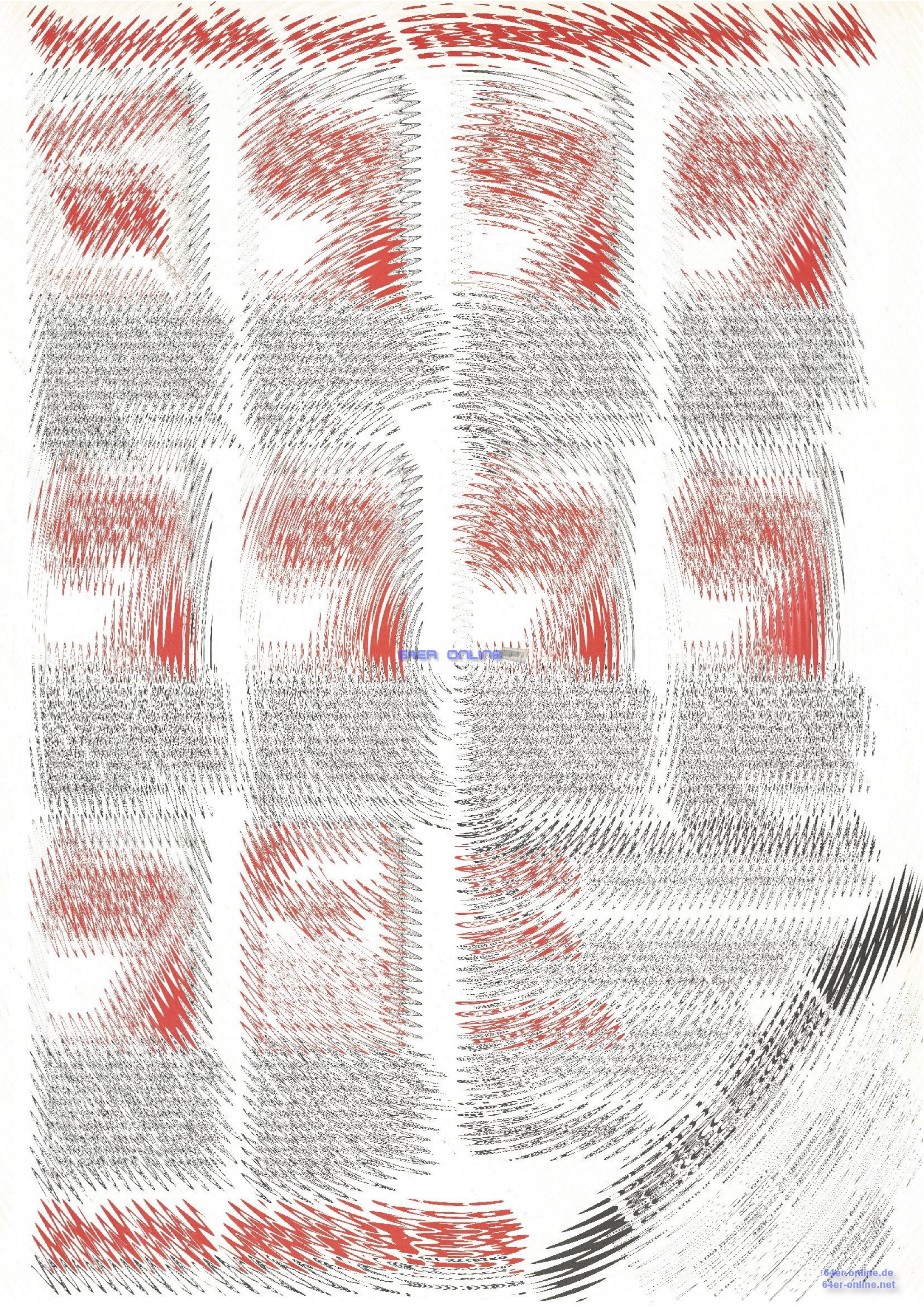
Die Wiederholung eines Programmausschnitts wird durch die For-Anweisung erreicht. Vor Betreten der Schleife muß bereits bekannt sein, wie oft die Schleife durchlaufen werden soll. Der Wert des Schleifenindexes darf also nicht in der Schleife selbst berechnet werden! In Listing 3 wird innerhalb einer Schleife die Fakultät einer positiven ganzen Zahl berech-

```
program intervall;
(* es wird ueberprueft, ob ein
Eingabewert innerhalb eines
Intervalls liegt *)
(* a <= x <= b *)
const a= -10.0;
      b=  20.0;
var x:real;
begin
writeln('Bitte Eingabe');
read(x);
if x>b then writeln('x: ',x:5:2,' groesser als obere Schranke')
else
begin
  if x>=a then writeln('x: ',x:5:2,' innerhalb der Schranke')
  else
    writeln('x: ',x:5:2,' kleiner als untere Schranke')
end
end.
```

Listing 1. Beispiel zur If-Then-Else-Klausel

```
program fall;
var note:integer;
    buchstabe:char;
begin
  read(note);
  if (note>=1) and (note<=6) then
    case note of
     1: writeln('sehr gut');
     2,3,4: writeln('o.k.');
    5,6: writeln('schwach')
    end
  else writeln('liegt ausserhalb des Bereichs')
end.
```

Listing 2. Die Case-Anweisung wählt unter mehreren Möglichkeiten


64ER ONLINE

net. Die Laufvariable (im Beispiel i) muß vom Typ Integer, Boolean, Char oder Aufzählungstyp sein. Anfangs- und Endwert der Schleifenvariablen können Konstanten, Variablen oder Ausdrücke desselben Typs sein. Die Laufvariable zählt die Anzahl der Wiederholungen. Vor jedem Durchlauf wird sie mit dem Endwert verglichen. Es wird überprüft, ob bereits genug Wiederholungen ausgeführt worden sind.

In der folgenden allgemeinen Form der For-Anweisung nennen wir die Laufvariable »vor«. Der Anfangswert heißt »a-Wert«. »e-Wert bezeichnet den Endwert.

```
FOR var := a-wert TO e-wert DO
aktion
```

oder

```
FOR var := a-wert DOWNTO
e-wert DO aktion
```

Nach jeder Wiederholung wird der Anfangswert entweder um eins erhöht (to) oder vermindert (downto). Eine Step-Anweisung wie in Basic gibt es in Pascal nicht.

Bei dem Beispiel in Listing 3 wird zunächst überprüft, ob der Eingabewert gleich Null ist. In diesem Fall steht das Ergebnis bereits fest. Bei einem Eingabewert größer Null wird die Fakultät berechnet und mit allen Zwischenergebnissen ausgegeben. N-Fakultät bedeutet:
1 * 2 * .... * N-1 * N .
So ist beispielsweise die Fakultät für
N=4 : 1 * 2 * 3 * 4 = 24 .

Schleifen mit »while« oder mit »repeat«?

Wenn vor Beginn der Schleife noch nicht feststeht, wie oft die Anweisungen wiederholt werden sollen, dann hilft die While- oder die Repeat-Schleife dem Programmierer weiter.

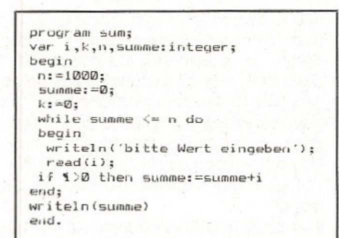
Solange eine bestimmte Bedingung erfüllt ist, wird bei der While-Schleife ein bestimmter Programmausschnitt ausgeführt. Eine While-Schleife ist folgendermaßen aufgebaut:

```
WHILE bed DO aktion
```

Der logische Ausdruck entscheidet, ob die Anweisung hinter dem »do« ausgeführt wird oder nicht. Ist das Ergebnis »true«, wird die Anweisung ausgeführt; bei dem Ergebnis »false« wird sie nicht ausgeführt. Die Anzahl der Wiederholungen hängt also von der Bedingung ab. Es ist möglich, daß der Ausdruck sofort den Wert »false« ergibt und die Anweisung damit überhaupt nicht ausgeführt wird. Es gibt auch Fälle, in denen die Schleife nicht beendet wird, weil der Ausdruck niemals den Wert »true« erhält. Einen solchen Fall nennt man eine unendliche Schleife. Diesen Fall sollte

```
program fakultaet;
var i,f,n:integer;
begin
 f:=1;
 writeln('bitte n eingeben');
 readln(n);
 if n=0 then writeln('Fakultaet = 1')
 else begin
  if n>0 then begin
   for i:=1 to n do begin
    f:=f*i;
    writeln(i:4,' Fakultaet: ',f:6);
   end
 end
 else writeln('negativ')
end;
end.
```

Listing 3. Die Fakultät-Funktion — mit der For-Schleife berechnet



```
program sum;
var i,k,n,summe:integer;
begin
 n:=1000;
 summe:=0;
 k:=0;
 while summe <= n do
 begin
  writeln('bitte Wert eingeben');
  read(i);
 if i>0 then summe:=summe+i
end;
writeln(summe)
end.
```

Listing 4. Beispielprogramm mit einer While-Schleife

```
program schleife;
var summe,k:integer;
begin
 summe:=0;
 k:=0;
 repeat
 k:=k+1;
 summe:=summe+k
until k>=10;
writeln('Summe: ',summe)
end.
```

Lisitng 5. Aufsummieren in einer Repeat-Anweisung

ein Programmierer auf jeden Fall vermeiden! Deshalb muß man auf diesen Fall achten, wenn das Ergebnis des logischen Ausdrucks auch innerhalb der Schleife geändert wird.

Das Beispiel in Listing 4 zeigt, wie innerhalb einer While-Schleife so lange Zahlen eingelesen und aufsummiert werden, bis die Zahl 1000 überschritten wird. Anhand dieses Beispiels nochmals die einzelnen Schritte bei While:

1. Der logische Ausdruck wird berechnet.
2. Falls der logische Ausdruck »true« ergibt, wird die Anweisung hinter Do ausgeführt — in diesem Fall eine Verbundanweisung. Eine Verbundanweisung wird immer wie eine einzelne Anweisung behandelt. Ist der logische Ausdruck »false«, so wird nicht diese Anweisung ausgeführt, sondern die darauf folgende. In diesem Fall heißt der Aufruf also:

```
writeln(summe);
```

3. Die Schritte 1 und 2 werden so lange ausgeführt, bis der logische Ausdruck »false« ist.

In Bild 2 werden die Teilschritte der While-Anweisung grafisch dargestellt.

Die Repeat-Anweisung ist in ei-

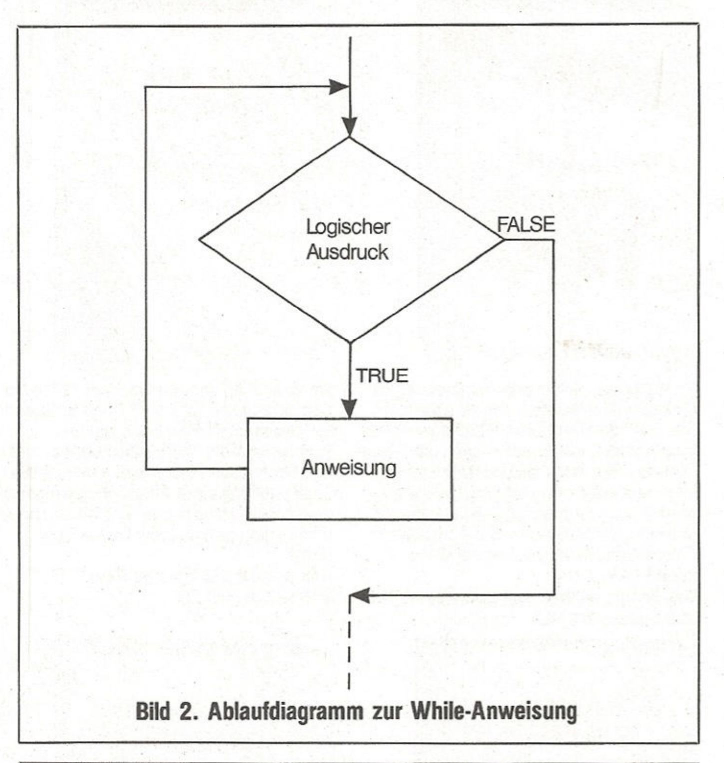


Bild 2. Ablaufdiagramm zur While-Anweisung

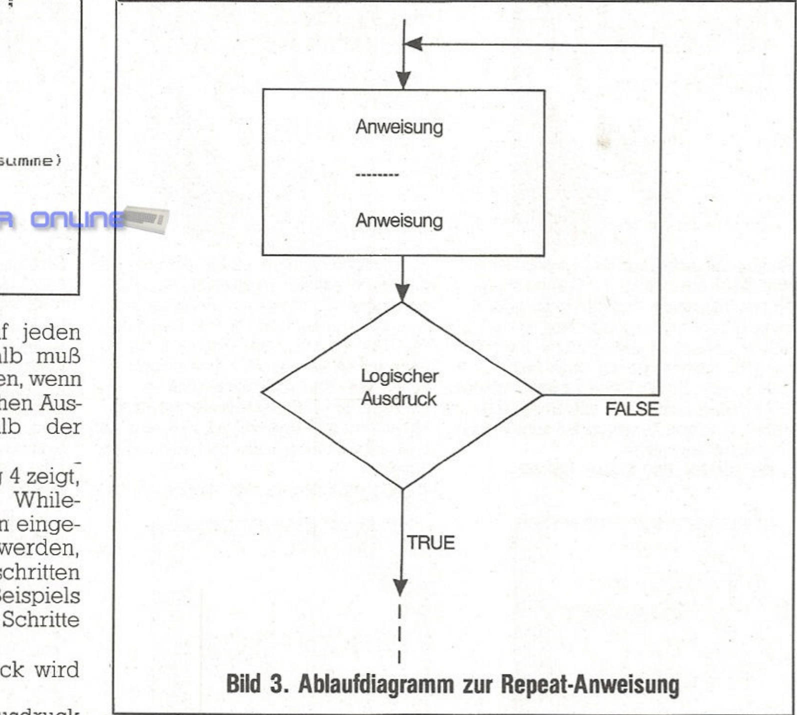


Bild 3. Ablaufdiagramm zur Repeat-Anweisung

WHILE	REPEAT
Test vor Schleifendurchlauf	Test nach Schleifendurchlauf
Test auf Fortsetzung der Schleife	Test auf Abbruch der Schleife
Wiederholungen, solange die Bedingung erfüllt ist	Wiederholungen, bis eine Bedingung erfüllt ist

Tabelle 1. Die Unterschiede zwischen While und Repeat

nem gewissen Sinne die Umkehrung der While-Anweisung. Bei ihr werden die Wiederholungen so lange ausgeführt, bis eine Bedingung erfüllt ist. Die allgemeine Form der Repeat-Anweisung lautet:

```
REPEAT Anweisung1; ... ;
       AnweisungN
UNTIL logischer Ausdruck;
```

Wenn der logische Ausdruck den Wert »true« hat, dann ist das Schleifenende erreicht. Solange er »false« ist, wird die Schleife wiederholt. Bei der Repeat-Anweisung wird (im Gegensatz zu While!) die Anweisung mindestens einmal durchlaufen. Ein Unterschied besteht auch darin,

Fortsetzung auf Seite 141

Von Basic zu Assembler (Teil 3)

Wozu lassen sich Schleifen in Maschinensprache einsetzen? Anhand von Beispielen zeigen wir Ihnen, wie man den Grafikspeicher löscht und beschreibt oder einen Rahmen um den Bildschirm legt. Auch das Beschreiben eines Bildschirmfensters wird in dieser Folge erwähnt. Diese universell zu gebrauchenden Routinen können zum Aufbau einer Makrobibliothek verwendet werden.

Vier Anwendungen der Schleifenprogrammierung werden Sie in dieser Folge kennenlernen: Das Löschen des Grafikspeichers, das Beschreiben des Grafik-Farbspeichers, einen Rahmenaufbau um den Bildschirm und schließlich das Beschreiben eines Bildschirmfensters.

Dies ist gleichzeitig auch die erste Reaktion auf Ihre Fragen, die Sie uns zum Thema Assembler gesandt haben. Die am Ende der letzten Folge versprochene Vorstellung der BLTUC-Routine muß noch ein wenig warten. Noch eine technische Vorbemerkung: Alle Programme wurden mit einem Nachfolger des Hypra-Ass — dem Programm Top-Ass — geschrieben. Aus dem Top-Ass wurden aber nur die Optionen verwendet, die auch in Hypra-Ass enthalten sind. Einige Pseudo-Opcodes heißen etwas anders, die Bedeutung ist aber leicht zu erkennen. In den Kommentaren finden Sie den jeweiligen Befehl auch in der Hypra-Ass-Syntax.

Init
Start
Startadresse in Vektor $FA/FB
Y = 0
Job
Job ausführen z.B.
— JSR JOB oder
— LDA $FE
STA ($FA),Y
Steuerung
LSB ($FA) + 1
Überlauf ?
Nein
Ja
MSB ($FB) +1
LSB ($FA) in Akku und Vergleich mit LSB (Ende+1) ($FC) Flags werden gesetzt
MSB ($FB) in Akku und Subtraktion von MSB (Ende +1) ($FD)
Carry frei ?
Ja
Ausgang
Nein
Ende

Bild 1. Flußdiagramm einer Doppelschleife für beliebige Laufzahlen

In der letzten Folge hatten wir uns eine spezielle Form der Doppelschleife angesehen, die es möglich machte, auch von ganzen Seiten (Pages) abweichende Speicherbereiche zu bearbeiten. Als Beispiel hatten wir den Bildschirminhalt invertiert. Diese Doppelschleife soll in verallgemeinerter Form diesmal Verwendung finden. Bild 1 zeigt Ihnen ein Flußdiagramm dieser allgemeinen 16-Bit-Schleife:

Schon in dem Beispiel zur Invertierung hatten wir die Startadresse als Vektor in zwei Zeropage-Speicherstellen geschrieben. Nun wird auch die Endadresse als Vektor $FC/FD gespeichert. So braucht nur im Initialisierungsteil von Aufgabe zu Aufgabe eine Änderung vorgenommen zu werden. Noch allgemeiner kann die Doppelschleife gestaltet werden durch eine Änderung des Jobteils. Handelt es sich beispielsweise um die Aufgabe, bestimmte Speicherbereiche zu beschreiben, dann kann auch der einzuschreibende Wert in eine Zeropage-Speicherstelle gepackt werden (hier in $FE). Will man allerdings auch die Art des Jobs offenhalten, dann verwendet man lediglich Sprünge in Unterprogramme. Der Jobteil heißt dann nur noch:

JSR JOB

Unsere Aufgabe ist es dann, an der Stelle JOB jeweils das gebrauchte Unterprogramm bereitzuhalten. Allerdings sollten solche Schleifenformen nicht oft benutzt werden, denn die Sprünge ins Unterprogramm verbrauchen relativ viel Rechenzeit.

Grafik-Farbspeicher belegen

Zwei Fragen treten häufig auf, die das Löschen oder Neubeschreiben der Grafikspeicher betreffen. Beide sind mit ein- und derselben Doppelschleife lösbar. Sehen wir uns zunächst einmal die Sache mit dem Grafik-Farbspeicher an. Im allgemeinen verwendet ein C 64-Grafik-Programmierer eine Bit-Map, die bei 8192 startet und ein Farb-RAM, das anstelle des normalen Bildschirmes (also ab 1024 = $400) zu finden ist. Der Benutzer des C 128 hat die Bit-Map am gleichen Ort, aber dafür ein extra Grafik-Farb-RAM ab $1C00. Zwar hat das Basic 7.0 des C 128 allerlei nette Grafik-Befehle anzubieten: Wenn aber gewünscht wird, eine schon auf dem Bildschirm sichtbare Zeichnung mit anderen Farben zu zeigen, stehen beide (also C 64- und C 128-Benutzer) vor demselben Problem. Eine globale Farbänderung ist beim C 128 nämlich nur bei gleichzeitigem Löschen der Bit-Map möglich! Wie also ist die Aufgabe lösbar?

In beiden Fällen ist in 1000 Speicherstellen ein bestimmter Wert einzuschreiben: Beim C 64 von $400 bis $7E8, beim C 128 von $01C00 bis $01FE8. Der Farbcode, der einzutragen ist, hat in beiden Fällen denselben Aufbau: Das untere Nibble (also die Bits 0 bis 3) enthält den Code der Hintergrund-, das obere Nibble (Bits 4 bis 7) den der Zeichenfarbe. C 128-Benutzer müssen vom Farbcode jeweils noch eine 1 abziehen. Sei ZF die Zeichen- und HF die Hintergrundfarbe, dann folgt für den einzuschreibenden Code F:

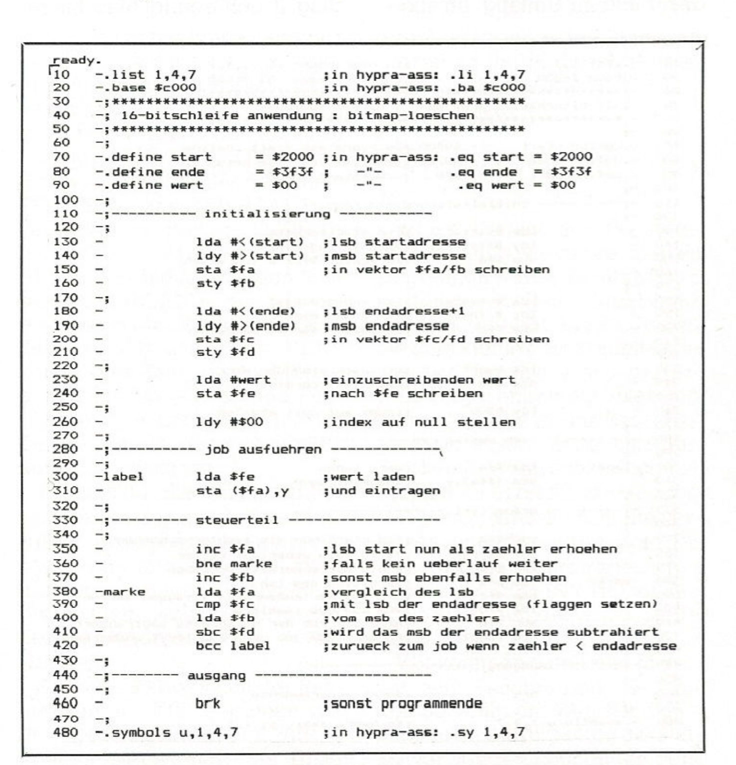
$F = 16 * ZF + HF$

Als Listing 1 finden Sie eine mögliche Lösung des Problems. Hier wurde in der Initialisierung (Zeilen 110 bis 270) auch die Belegung der Zeropage-Adressen mit dem Startwert ($FA/FB), dem Endwert ($FC/FD) und dem Farbcode ($FE) vorgenommen:

Zum Listing selbst muß sicher nichts mehr gesagt werden: Es ist ausführlich kommentiert und entspricht genau der oben im Flußdiagramm gezeigten Doppelschleife. Zum Starten dieses Maschinenprogrammes: Wenn Sie es einfach durch

```
SYS 49152
(C 128: BANK 0:SYS49152)
```

mit auf dem Bildschirm vorhandener Abbildung ablaufen lassen, wird der Farbcode $F0 eingetragen, also hellgrau auf schwarzen Hintergrund. Falls Sie im Textmodus sind, entspricht das dem Zeichen mit dem Code $F0 (dezimal 240, ein inverses Grafik-Zeichen), das nun den gesamten Bildschirm ziert. Möchten Sie eine andere Farbenkombination erzielen, dann haben Sie mehrere Möglichkeiten. Entweder lassen Sie einfach die Zeilen 230 und 240 wegfallen und POKEn vor dem Routinenaufruf Ihren Wert F in die Speicherstelle $FE oder



```
ready.
10   -.list 1,4,7                ;in hypra-ass: .li 1,4,7
20   -.base $c000                ;in hypra-ass: .ba $c000
30   -;**************************************************
40   -; 16-bitschleife anwendung : bitmap-loeschen
50   -;**************************************************
60   -;
70   -.define start      = $2000 ;in hypra-ass: .eq start = $2000
80   -.define ende       = $3f3f ;    -"-       .eq ende  = $3f3f
90   -.define wert       = $00   ;    -"-        .eq wert = $00
100  -;
110  -;---------- initialisierung -----------
120  -;
130  -            lda #<(start)  ;lsb startadresse
140  -            ldy #>(start)  ;msb startadresse
150  -            sta $fa        ;in vektor $fa/fb schreiben
160  -            sty $fb
170  -;
180  -            lda #<(ende)   ;lsb endadresse+1
190  -            ldy #>(ende)   ;msb endadresse
200  -            sta $fc        ;in vektor $fc/fd schreiben
210  -            sty $fd
220  -;
230  -            lda #wert      ;einzuschreibenden wert
240  -            sta $fe        ;nach $fe schreiben
250  -;
260  -            ldy #$00       ;index auf null stellen
270  -;
280  -;---------- job ausfuehren ------------
290  -;
300  -label       lda $fe        ;wert laden
310  -            sta ($fa),y    ;und eintragen
320  -;
330  -;--------- steuerteil -----------------
340  -;
350  -            inc $fa        ;lsb start nun als zaehler erhoehen
360  -            bne marke      ;falls kein ueberlauf weiter
370  -            inc $fb        ;sonst msb ebenfalls erhoehen
380  -marke       lda $fa        ;vergleich des lsb
390  -            cmp $fc        ;mit lsb der endadresse (flaggen setzen)
400  -            lda $fb        ;vom msb des zaehlers
410  -            sbc $fd        ;wird das msb der endadresse subtrahiert
420  -            bcc label      ;zurueck zum job wenn zaehler < endadresse
430  -;
440  -;-------- ausgang ---------------------
450  -;
460  -            brk            ;sonst programmende
470  -;
480  -.symbols u,1,4,7            ;in hypra-ass: .sy 1,4,7
```

Listing 1. Ein Programm zum Beschreiben des Grafik-Farb-RAM

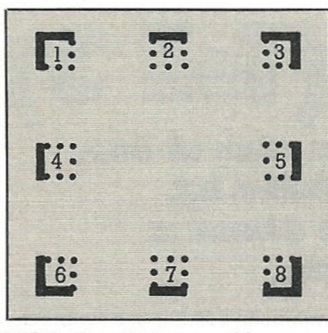


Bild 2. Zum Rahmenproblem: Welche Zeichen werden benötigt?

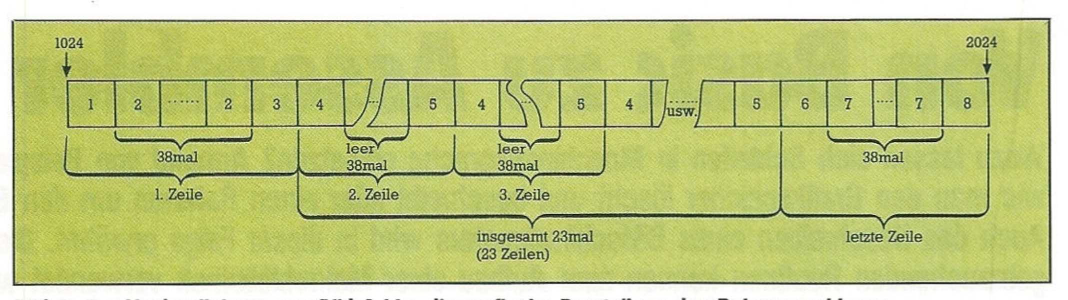


Bild 3. Zur Verdeutlichung von Bild 2 hier die grafische Darstellung des Rahmenproblems

aber Sie verändern das Maschinenprogramm vor dem Aufruf, indem Sie an die Stelle in Zeile 230, an der der WERT steht, Ihr F POKEn. Das ist (vorausgesetzt, Sie belassen den Start bei $C000) die Speicherzelle $C011 (das ist dezimal 49169). Ein Aufruf könnte dann beispielsweise so aussehen:

```
10 INPUT"ZF,HF =";ZF,HF
20 F = 16*ZF + HF
30 POKE 49169,F
40 SYS 49152
```

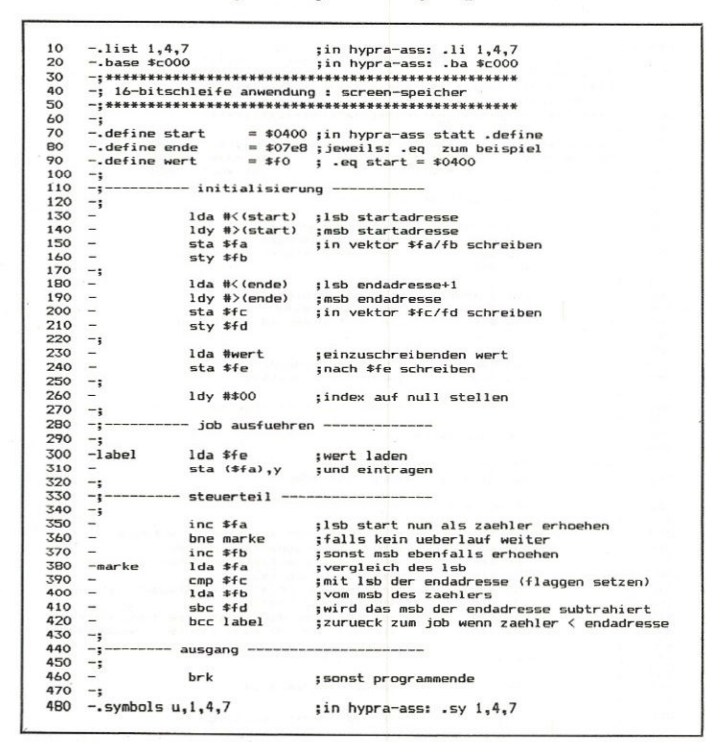
Auf ähnliche Weise können natürlich auch die Start- und/oder Endadressen variiert werden, so daß beispielsweise nur der halbe Farbspeicher neu belegt wird. C 128-Benutzer müssen zum Ändern des Grafik-Farb-RAM die Werte in den Zeilen 70 und 80 auf die oben genannten Adressen einstellen.

Bit-Map löschen

Vom eben gezeigten Beispiel zum Löschen einer Bit-Map ist es nur ein kleiner Schritt. Lediglich der Ort einer Bit-Map ist ein anderer und ihr Umfang. Im allgemeinen startet der Grafik-Speicher bei 8192 ($2000), hat eine Ausdehnung von 8000 Bytes (nämlich 200*320/8) und endet bei $3F3F (dezimal 16191). Der einzutragende Wert ist Null. Listing 1 unterscheidet sich vom Listing 2 somit nur durch die Zeilen 70 bis 90.

Daran können Sie ersehen, wie vielseitig unsere Doppelschleife ist. C 128-Benutzer werden nun meinen, daß sie das Listing 2 und — falls ihnen niemals das Problem der Farbänderung eines fertigen Bildes unterkommt — auch das Listing 1 entbehren könnten. Sollten sie aber jemals in die Lage kommen, eine andere Bit-Map und einen anderen Grafik-Farbspeicher als die softwaremäßig voreingestellten bedienen zu wollen, dann sind sie in der gleichen Lage wie ein C 64-Benutzer. Alle Grafik-Kommandos zielen nur auf die oben erwähnten Standard-Grafik-Speicherbereiche. Jede andere Bit-Map und jedes andere Farb-RAM muß mit selbst gebastelten Befehlssequenzen behandelt werden.

Noch eine kleine Bemerkung am Rande: Falls es Ihnen in den Sinn kommen sollte, das zum Listing 2 gehörende Maschinenprogramm einfach mal zu starten, dann bedenken Sie, daß durch Hypra-Ass im fraglichen — zu löschenden — Speicherbereich Quelltexte und Label abgelegt sind. Speichern Sie sie also vor dem Start ab! Noch schlimmer ergeht es C 128-Benutzern, die den Top-Ass verwenden: Das Löschen des Grafik-Bereiches reißt ein tiefes Loch ins Top-Ass-Programm. Top-Ass und normale Grafik können nicht gleichzeitig betrieben werden.



```
10   -.list 1,4,7                  ;in hypra-ass: .li 1,4,7
20   -.base $c000                  ;in hypra-ass: .ba $c000
30   -;*********************************************
40   -; 16-bitschleife anwendung : screen-speicher
50   -;*********************************************
60   -;
70   -.define start      = $0400 ;in hypra-ass statt .define
80   -.define ende       = $07e8 ;jeweils: .eq  zum beispiel
90   -.define wert       = $f0   ; .eq start = $0400
100  -;
110  -;---------- initialisierung -----------
120  -;
130  -            lda #<(start)  ;lsb startadresse
140  -            ldy #>(start)  ;msb startadresse
150  -            sta $fa        ;in vektor $fa/fb schreiben
160  -            sty $fb
170  -;
180  -            lda #<(ende)   ;lsb endadresse+1
190  -            ldy #>(ende)   ;msb endadresse
200  -            sta $fc        ;in vektor $fc/fd schreiben
210  -            sty $fd
220  -;
230  -            lda #wert      ;einzuschreibenden wert
240  -            sta $fe        ;nach $fe schreiben
250  -;
260  -            ldy #$00       ;index auf null stellen
270  -;
280  -;---------- job ausfuehren ------------
290  -;
300  -label       lda $fe        ;wert laden
310  -            sta ($fa),y    ;und eintragen
320  -;
330  -;--------- steuerteil -----------------
340  -;
350  -            inc $fa        ;lsb start nun als zaehler erhoehen
360  -            bne marke      ;falls kein ueberlauf weiter
370  -            inc $fb        ;sonst msb ebenfalls erhoehen
380  -marke       lda $fa        ;vergleich des lsb
390  -            cmp $fc        ;mit lsb der endadresse (flaggen setzen)
400  -            lda $fb        ;vom msb des zaehlers
410  -            sbc $fd        ;wird das msb der endadresse subtrahiert
420  -            bcc label      ;zurueck zum job wenn zaehler < endadresse
430  -;
440  -;-------- ausgang --------------------
450  -;
460  -            brk            ;sonst programmende
470  -;
480  -.symbols u,1,4,7              ;in hypra-ass: .sy 1,4,7
```

Listing 2. Blitzartiges Löschen der Bit-Map

Bildschirm umrahmen

Viele Leser wollten wissen, wie man in Assembler einen Rahmen um den Textbildschirm legen kann. Gehen wir also dieses Problem möglichst allgemeingültig an. Wir brauchen dazu acht verschiedene Grafikzeichen (siehe dazu das Bild 2).

Außerdem erkennen wir recht schnell, daß wir bei diesem Problem andere Schleifen benutzen müssen. In Bild 3 ist die Aufgabe grafisch aufgeschlüsselt.

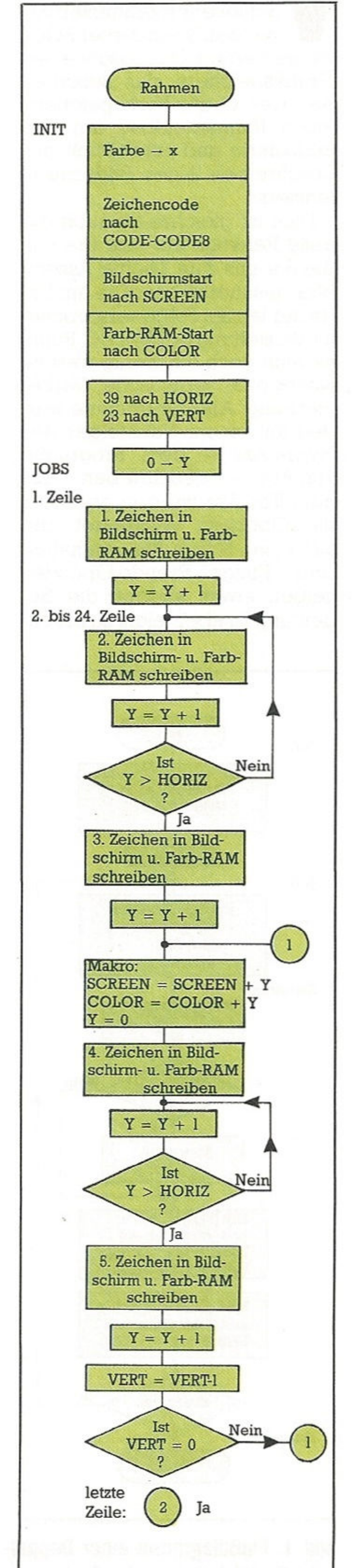
Die Nummern entsprechen jeweils den verschiedenen Zeichen, die in eine Speicherstelle einzuschreiben sind. Es handelt sich um folgende:

Nr.	Hex.	Dez.	Beschreibung
1	4F	79	Winkel links oben
2	77	119	Linie oben
3	50	80	Winkel rechts oben
4	74	116	Linie linksseitig
5	6A	106	Linie rechtsseitig
6	4C	76	Winkel links unten
7	6F	111	Linie unten
8	7A	122	Winkel rechts unten

Im Listing 3 sind diese acht Zeichen in der Initialisierungsphase in acht Zeropage-Speicherstellen geschrieben worden ($1B bis $22), die CODE1 bis CODE8 genannt wurden.

Bevor wir dieses Programm besprechen, gebe ich Ihnen in Bild 4 noch das Flußdiagramm dazu an.

Das Programm ist in drei Teilen angeordnet: Die Initialisierung, in der alle Startwerte festgelegt werden. Sie liegen durchweg in Zeropage-Speicherstellen, was durch einfaches Ändern dieses Programmteils das Programm flexibel hält. Wir hätten noch einen Schritt weiter gehen können, indem wir





```
10   -.list 1,4                  ;in hypra-ass: .li 1,4
20   -.base $c000                ;in hypra-ass: .ba $c000
30   -;*************************************
40   -; verschachtelte schleifen  anwendung
50   -; allgemeiner bildschirmrahmen
60   -;*************************************
70   -;
80   -;lagerplatz fuer die zeichen: (hpyra-ass jeweils: .eq code1 = $1b usw.)
90   -.define code1      = $1b   ;zeichen O = 4f
100  -.define code2      = $1c   ;zeichen 7 = 77
110  -.define code3      = $1d   ;zeichen P = 50
120  -.define code4      = $1e   ;zeichen % = 74
130  -.define code5      = $1f   ;zeichen ' = 6a
140  -.define code6      = $20   ;zeichen L = 4c
150  -.define code7      = $21   ;zeichen / = 6f
160  -.define code8      = $22   ;zeichen ; = 7a
170  -;vektoren fuer bildchirm- und farb-ram:
180  -.define screen     = $fb   ;bildschirmstart
190  -.define color      = $fd   ;farbramstart
200  -;zaehler:
210  -.define horiz      = $23   ;zaehler fuer horizontale
220  -.define veri       = $24   ;zaehler fuer vertikale
230  -;der farbcode wird nur im x-register gespeichert
240  -;
250  -;definition eines makro: aktuell
260  -;in hypra-ass stattdessen: .ma aktuell(screen,color)
270  -.macro aktuell(screen,color)
280  -             clc
290  -             tya
300  -             adc screen
310  -             sta screen
320  -             lda screen+1
330  -             adc #$00
340  -             sta screen+1
350  -             clc
360  -             tya
370  -             adc color
380  -             sta color
390  -             lda color+1
400  -             adc #$00
410  -             sta color+1
420  -             ldy #$00
430  -.macend                     ;in hypra-ass: .rt
440  -;
450  -;nun gehts los:
460  -;
470  -;-------------- initialisierung ------------
480  -;
490  -             ldx #$02        ;farbcode ins x-register
500  -             lda #$4f        ;alle zeichen in die speicher packen
510  -             sta code1
520  -             lda #$77
530  -             sta code2
540  -             lda #$50
550  -             sta code3
560  -             lda #$74
570  -             sta code4
580  -             lda #$6a
590  -             sta code5
600  -             lda #$4c
610  -             sta code6
620  -             lda #$6f
630  -             sta code7
640  -             lda #$7a
650  -             asl bild        ;=  *16
660  -             rol bild+1      ;zur sicherheit carry in msb rotieren
670  -             asl bild        ;=  *32 (zweiter summand)
680  -             rol bild+1      ;nochmal carry ins msb rotieren
690  -             lda zwsp        ;addition beider summanden (carry ist frei)
700  -             adc bild        ;lsb von 32*z1
710  -             sta bild        ;lsb von 40*z1
720  -             lda bild+1      ;msb von 32*z1
730  -             adc #$00        ;eventuell carry addieren
740  -             sta bild+1      ;im vektor bild steht nun 40*(z1-1)
750  -;
760  -             clc             ;addition von s1
770  -             lda bild        ;lsb
780  -             adc spalte      ;das ist ja s1
790  -             sta bild
800  -             lda bild+1      ;msb
810  -             adc #$00        ;eventuell carry addieren
820  -             sta bild+1      ;im vektor bild steht nun 40*(z1-1)+s1
830  -;
840  -             clc             ;farbramadresse berechnen
850  -             lda #<(colram)  ;lsb
860  -             adc bild
870  -             sta color       ;vektor aktuelle farbramadresse lsb
880  -             lda #>(colram)  ;msb
890  -             adc bild+1
900  -             sta color+1     ;vektor msb
910  -;
920  -             clc             ;aktuelle bildschirmadresse berechnen
930  -             lda #<(screen)  ;lsb
940  -             adc bild
950  -             sta bild        ;vektor bildschirm lsb
960  -             lda #>(screen)  ;msb
970  -             adc bild+1
980  -             sta bild+1      ;bildschirmvektor msb
990  -;
1000 -marke5       ldx #farbe      ;durch poke veraenderbar
1010 -;
1020 -;******* hier beginnen die verschachtelten schleifen *******
1030 -label1       ldy #$00        ;spaltenzaehler zuruecksetzen
1040 -label2       lda #wert       ;durch poke veraenderbar
1050 -             sta (bild),y    ;auf bildschirm schreiben
1060 -             txa             ;farbwert umladen
1070 -             sta (color),y   ;und in farbram schreiben
1080 -             iny
1090 -             cpy spdiff      ;zeile fertig ?
1100 -             bcc label2      ;noch nicht, dann weitermachen
1110 -;ansonsten werden zu den vektoren bild und color 40 addiert:
1120 -             clc             ;bild aktualisieren
1130 -             lda bild        ;lsb
1140 -             adc #$28        ;das ist dezimal 40
1150 -             sta bild
1160 -             lda bild+1      ;msb
1170 -             adc #$00        ;carry addieren
1180 -             sta bild+1      ;vektor bild ist aktualisiert
1190 -;
1200 -             clc             ;color aktualisieren
1210 -             lda color       ;lsb
1220 -             adc #$28        ;wieder 40 dazuzaehlen
1230 -             sta color
1240 -             lda color+1     ;msb
1250 -             adc #$00        ;addieren des carry
1260 -             sta color+1     ;vektor color ist aktualisiert
1270 -;
1280 -             dec zdiff       ;zeilenzaehler herunterzaehlen
1290 -             bne label1      ;wenn noch > 0, naechste zeile schreiben
1300 -;
1310 -;im anderen fall ist die aufgabe erledigt:
1320 -             rts             ;zurueck zum aufrufenden programm oder basic
1330 -;
1340 -.symbols u,1,4,7             ;in hypra-ass: .sy 1,4,7
```

Listing 5. Das Assembler-Programm »Teilbereich«



Zeile 80 statt des DEC ("3FF") einfach 1023 ein.

Natürlich funktioniert das so einwandfrei, aber es dauert! In Assembler geht das Ausfüllen auch dann noch blitzartig, wenn wir als Rechteck den gesamten Bildschirm vorgeben. Das einzige, was Kopfzerbrechen bereiten kann, ist die Berechnung der Startadresse. Aber auch dazu gibt es einen schnellen Weg, wie Sie gleich noch sehen werden. Bild 7 gibt Ihnen als Anhaltspunkt ein Flußdiagramm zum Assemblerprogramm TEILBEREICH.

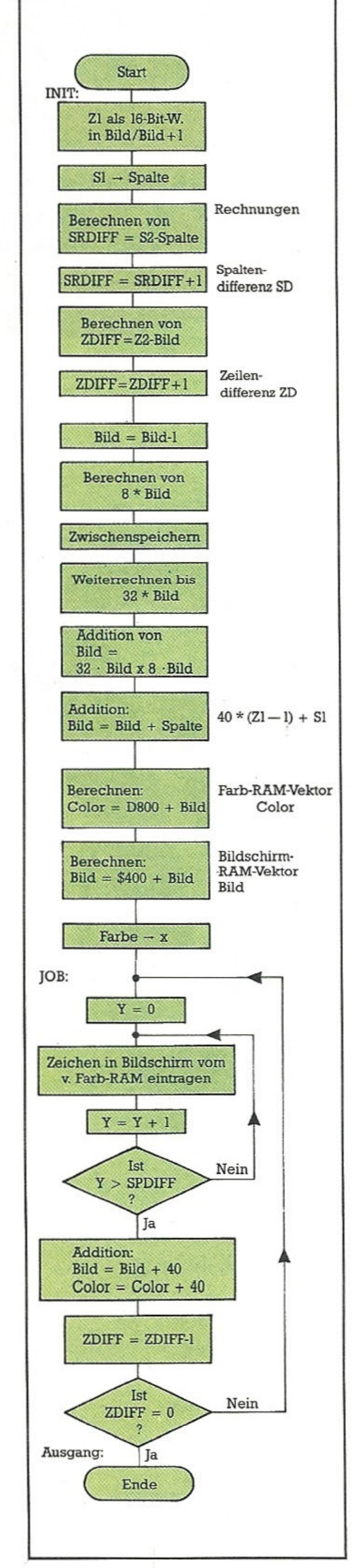


Bild 7. Flußdiagramm zum Programm »Teilbereich«

Listing 5 schließlich enthält unser gleich zu besprechendes Programm »Teilbereich«.

Im ersten Teil des Listings sehen Sie wieder die Definitionen, die der Assembler beim Assemblieren statt der Merktexte einsetzt. Diesmal sind von Zeile 130 bis 180 Werte vorgegeben (purpurfarbene Zeichen A werden in das Rechteck geschrieben, das von Spalte 4/Zeile 3 bis Spalte 15/ Zeile 13 reicht), von 220 bis 270 Zeropage-Speicherstellen, die der Speicherung der Spaltendifferenz (SPDIFF), Zeilendifferenz (ZDIFF), der Spalte Sl (SPALTE) und eines Zwischenwertes der Rechnung (ZWSP) dienen. Außerdem sind zwei Vektoren vorgesehen: BILD (für die laufende Bildschirmspeicherposition) und COLOR (dasselbe für das Farb-RAM). Die Zeilen 310 und 320 enthalten die Festlegung der Basisadressen des Bildschirmspeichers und des Farb-RAMs (jeweils -1).

Der erste Teil des eigentlichen Programmes dient der Initialisierung und der Berechnung der Steuergrößen (im Basic-Programm waren das SD, ZD und Sl). Zunächst speichern wir Zl als 16-Bit-Wert im Vektor BILD. Dazu muß das MSB von BILD (also BILD+1) noch auf Null gesetzt werden, denn Zl ist ja nur ein 8-Bit-Wert. Der Wert Sl wird danach in SPALTE geschrieben (Zeilen 430 und 440). Nun fängt die Rechnerei an. Wir bilden die Spalten- und die Zeilendifferenzen jeweils durch einfache 8-Bit-Subtraktionen. Zur Erinnerung: Vor einer normalen Subtraktion muß immer das Carry-Bit gesetzt werden, denn es dient gewissermaßen als »Stelle, die geborgt werden kann«. Die Wirklichkeit ist etwas komplexer, aber das können Sie — falls es Sie interessiert — im Assemblerkurs oder einschlägigen Lehrbüchern nachlesen. Nach vollendeter Subtraktion werden SPDIFF und ZDIFF jeweils noch durch den INC-Befehl um 1 erhöht. Das hat seinen Grund in den Schleifenabbruchbedingungen im 2. Teil unseres Programmes. Soweit war die Sache noch recht einfach. Nun kommt aber die Berechnung des Startpunktes im Bildschirm- und im Farb-RAM. Das heißt wir haben den Ausdruck

40*(Zl-1)+Sl

zu berechnen. Wir bedienen uns in der Hauptsache der Assembler-Befehle ASL und ROL dazu, die — weil wir sie recht selten erleben — hier nochmal kurz erläutert werden sollen. Die Bilder 8 und 9 illustrieren die Erklärungen.

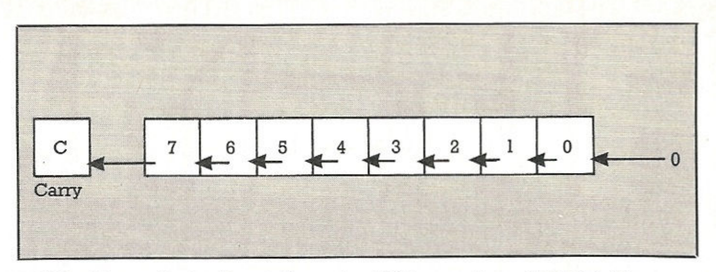


Bild 8. Die grafische Darstellung der Wirkung eines ASL-Befehls

ASL (arithmetic shift left = arithmetisches Linksverschieben) schiebt jedes Bit des angesprochenen Bytes um eine Position nach links. An die Stelle des Bit 0 tritt eine Null und das Bit 7 landet im Carry. Was das bedeuten kann, sehen wir uns an einer Zahl im Dezimalsystem mal an:

240 unsere Zahl

2400 nun ist jede Stelle einmal nach links verschoben

Sie sehen, daß das einer Multiplikation um den Faktor 10 entspricht. Im binären Zahlensystem, das unser Computer verwendet, ist die Zahlenbasis nicht mehr 10, sondern 2. Deshalb führt hier jede Linksverschiebung zu einer Multiplikation mit 2. Sehen wir uns das am konkreten Beispiel an: Der höchste Wert für Zl in unserem Programm ist 25 (es gibt nur 25 Zeilen). Wenn wir von Zl-1 ausgehen ist das 24:

0001 1000 unsere Zahl 24
1.ASL: 0011 0000 das ist 2*24
2.ASL: 0110 0000 das ist 4*24
3.ASL: 1100 0000 das ist 8*24

Vorsicht! Wenn wir nun nochmal die ASL-Operation anwenden, schieben wir das Bit 7 in das Carry-Bit. Das Ergebnis ist für eine 8-Bit-Zahl zu groß geworden, eine 16-Bit-Zahl ist vonnöten. Wir müßten dann die 1 aus dem Carry-Bit als Bit 0 des MSB verwenden können. Das wird mit ROL (rotate left = nach links rotieren) möglich. Auch verschiebt der Befehl jedes Bit um eine Stelle nach links. Als neues Bit 0 aber tritt der Inhalt des Carry-Bit ein. Parallel dazu wandert das Bit 7 ins Carry-Bit. Der weitere Weg in unserem Beispiel muß nun also lauten:

4.ASL: 1 1000 0000 und dann
1.ROL: 0000 0001 1000 0000
MSB LSB

Das ist nun 16*24. Durch ASL wurde Bit 7 ins Carry geschoben (deshalb steht es oben links außen), durch ROL (bezogen auf die Speicherstelle, die als MSB eingesetzt wird) wandert es dann ins MSB als Bit 0. Die dazugehörige Befehlssequenz lautet (siehe beispielsweise Zeilen 650/660 des Listing 5):

ASL BILD
ROL BILD+1

Vermutlich werden Sie sich nun fragen, was das alles mit unserem Problem zu tun hat, 40*(Zl-1)+Sl zu berechnen. Sehr viel! Rechnen Sie es mal nach. Es gilt nämlich:

40*(Zl-1) = 8*(Zl-1) + 32*(Zl-1)

Sowohl 8 als auch 32 sind aber Potenzen von 2, das bedeutet, daß sie durch mehrmaliges Multiplizieren der 2 mit sich selbst herauskommen: 2↑3 = 8 und 2↑5 = 32. Es ist damit möglich, durch mehrmaliges Anwenden der ASL-Operation auf (Zl-1) beide Summanden zu erzeugen. Die Addition ergibt dann unseren gewünschten Wert 40*(Zl-1). Wir müssen nur noch Sl hinzurechnen. Wie wir vorhin noch sehen konnten, dürfen wir die drei ersten ASL-Operationen noch ohne Rücksicht auf Verluste durchführen. Danach allerdings muß jedem ASL das dazugehörige ROL folgen, denn das Ergebnis kann eine 16-Bit-Zahl sein.

In unserem Programm beginnen wir diese Berechnung in Zeile 580. Dort wird zunächst sicherheitshalber das Carry-Bit-

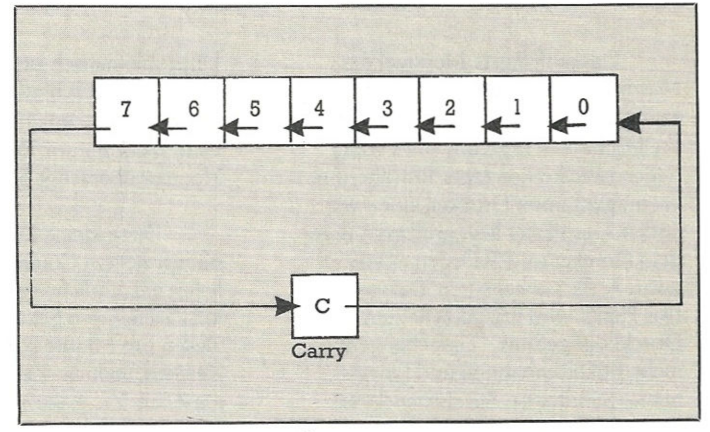


Bild 9. Die grafische Darstellung der Wirkung des ROL-Kommandos



```
10 REM ******** AUFRUFPROGRAMM FUER TEILBILDSCHIRM
 **********
20 REM IN REM-BEMERKUNGEN JEWEILS WERTE FUER DEN C
128
30 REM VORAUSSETZUNG, DASS BEIM C64 AB $C000
40 REM                     BEIM C128 AB $1400 DAS
PROGRAMM LIEGT.
50 REM
60 REM ------- EINGABE DER PARAMETER ---------
70 PRINT CHR$(147):INPUT"LINKE OBERE ECKE Z1,S1";Z
1,S1
80 INPUT"RECHTE UNTERE ECKE Z2,S2";Z2,S2
90 INPUT"FARBE,WERT";F,W
100 REM ------- EINPOKEN DER WERTE ------------
110 POKE 49157,Z1:REM POKE 5125,Z1
120 POKE 49161,S1:REM POKE 5129,S1
130 POKE 49166,S2:REM POKE 5134,S2
140 POKE 49175,Z2:REM POKE 5143,Z2
150 POKE 49255,F:REM POKE 5223,F
160 POKE 49259,W:REM POKE 5227,W
170 REM ------ AUFRUF DER ROUTINE -----------
180 SYS49152
190 END
```

Listing 6. Ein Basic-Programm, das alle POKE-Befehle ausführt und unsere Routine ansteuert

gelöscht für das erste ROL-Kommando. Durch DEC BILD erzeugen wir den Wert Z1-1 (nach BILD hatten wir ja Z1 geschrieben, BILD+1 wurde reserviert für das MSB und eine Null eingetragen). Dreimaliges ASL BILD erzeugt 8*(Z1-1). Sie erinnern sich: Erst vom vierten ASL an mußte beim größten erlaubten Z1-Wert auch ROL eingesetzt werden. In den Zeilen 640/650 packen wir diesen Summanden in den Zwischenspeicher. Danach sind noch zwei ASL mit anschließenden ROL-Kommandos nötig, um in BILD/BILD+1 den Wert 32*(Z1-1) zu erzeugen. Von Zeile 690 bis 740 addieren wir beide Summanden und erhalten so 40*(Z1-1). Normalerweise muß vor jeder Addition mit ADC (denn das heißt ja »add with carry«, also addiere mit dem Carry-Bit, sogar addiere auch das Carry-Bit) das Carry-Bit durch CLC freigemacht werden. Hier können wir uns das aber ersparen, weil wir wissen, daß auch der größte Z1-Wert bei der letzten ROL-Operation nur eine Null ins Carry-Bit schieben konnte. Bei der folgenden Addition von S1 (Zeilen 760 bis 820) entsteht schließlich unser gewünschter Startwert — aber noch ohne Berücksichtigung der Anfangsadressen von Bildschirm- und Farb-RAM. Das kommt nun: Von Zeile 840 bis 900 entsteht im Vektor COLOR/COLOR+1 der Zeiger auf die erste Farb-RAM-Stelle, von Zeile 920 bis 980 im Vektor BILD/BILD+1 schließlich der auf die erste Bildschirmspeicherstelle. Zum Abschluß dieser Programmphase schieben wir noch den Farbcode ins X-Register.

Nun kommt die Doppelschleife, die den Bildschirmausschnitt beschreibt. Das Y-Register dient als Zähler jeweils einer Zeile. Der Zeichencode gelangt — ebenso wie der Farbcode — nach dem gleichen Rezept in die notwendigen Speicherstellen, das wir auch schon in den vorangegangenen Programmbeispielen gezeigt haben (das war die Sequenz, die wir dort auch als Makro hätten schreiben können: Hier in den Zeilen 1050 bis 1080). Jeweils am Ende der kleinen Schleife (in 1090 und 1100) wird der aktuelle Y-Zählerstand verglichen mit der Spaltendifferenz (SPDIFF) und — falls die Zeile noch nicht fertig ist — der nächste Durchlauf eingeleitet. Andernfalls addieren wir sowohl zum Vektor BILD/BILD+1 als auch zu COLOR/COLOR+1 den Wert 40, rücken also eine Zeile runter. In 1280 dient ein DEC ZDIFF dazu, die Anzahl bearbeiteter Zeilen abzustreichen, bis dabei eine Null auftritt. Ist das noch nicht der Fall, wird zurückverzweigt nach LABEL1, wo wir den Y-Zähler neu initialisieren.

Der letzte Teil unseres Programmes besteht lediglich wieder aus einem lapidaren RTS.

Wenn Sie unser Programm nach dem Assemblieren einfach mittels SYS 49152 starten, sind automatisch die Beispielwerte eingesetzt. C 128-Benutzern sei wieder geraten, das Programm nach $1400 zu legen, damit auch der Farbspeicher belegt werden kann. Hier erfolgt der Start dann durch

BANK15:SYS DEC("1400").

Sehr flexibel wird »Teilbildschirm« aber erst durch einige POKE-Kommandos ins Programm hinein. Folgende Speicherstellen werden dann angesteuert:

Parameter	Ort im Programm	POKE-Adressen C 64	POKE-Adressen C 128
Z1	MARKE1+1	49157	5125
S1	MARKE2+1	49161	5129
S2	MARKE3+1	49166	5134
Z2	MARKE4+1	49175	5143
Farbe	MARKE5+1	49255	5223
Wert	LABEL2+1	49259	5227

Vorausgesetzt wird bei diesen Angaben, daß im C 64 ab $C000 und im C 128 ab $01400 unser Programm zu finden ist. Listing 6 zeigt Ihnen, wie ein Basic-Programm aussehen kann, das alle POKE-Befehle ausführt und unsere Routine ansteuert. Falls geplant ist, Benutzer an dieses Programm zu lassen, die es nicht kennen, sollten Sie auch noch eine Überprüfung der Eingabewerte einbauen, andernfalls könnte beispielsweise Unsinn herauskommen, wenn S2 kleiner als S1 angegeben wird oder mal eine Zeilennummer, die größer als 25 ist. Die erlaubten Werte sind jeweils Spaltenwerte zwischen 1 und 40, Zeilen zwischen 1 und 25.

Wie man Schleifen in Assembler anwenden kann und auf wie verschiedene Arten das möglich ist, haben Ihnen diese vier Beispiele sicher zeigen können. Aber mit Schleifen ist noch viel mehr machbar.

Damit werden wir uns in den nächsten Folgen befassen.

(Heimo Ponnath/dm)

Fortsetzung von Seite 132

daß zwischen Repeat und Until mehrere Anweisungen stehen dürfen (in der While-Schleife muß man eine Verbundanweisung verwenden), da diese Schlüsselwörter die gleiche Wirkung wie Begin und End haben.

Listing 5 zeigt die Anwendung der Repeat-Anweisung. Die Zahlen 1 bis 10 werden innerhalb der Schleife addiert. Zur Verdeutlichung nochmals die einzelnen Schritte:

1. Der Schleifenkörper, das sind die Anweisungen zwischen Repeat und Until, wird ausgeführt.
2. Der logische Ausdruck wird berechnet.
3. Hat der Ausdruck den Wert »true«, ist das Ende der Schleife erreicht. Die nächste Anweisung wird ausgeführt.

Hat der Ausdruck den Wert »false«, so wird die Schleife nochmals durchlaufen.

4. Die Schritte 1 bis 3 werden so oft ausgeführt, bis der logische Ausdruck den Wert »true« annimmt.

In Bild 3 werden diese Schritte nochmals grafisch dargestellt. Wie bei der Verwendung der While-Anweisung muß der Programmierer auch bei der Repeat-Schleife darauf achten, daß er den Wert des Schleifenindexes innerhalb der Schleife nicht verändert. Die Unterschiede zwischen While und Repeat haben wir in Tabelle 1 nochmals zusammengefaßt.

Damit sind alle Anweisungen mit Ausnahme der With-Anweisung vorgestellt. Diese Anweisung werden wir zusammen mit dem Datentyp Record in der übernächsten Folge unseres Kurses besprechen. In der nächsten Folge beschäftigen wir uns mit Ausdrücken und Standardfunktionen und zeigen, wie man sich in Pascal eigene Dateitypen definieren kann. Und außerdem werden die Ausschnitts- und Aufzählungstypen behandelt.

In den weiteren Folgen dieses Kurses stellen wir vor: die Datentypen Array und Record, dann Set (Menge) und Pointer (Zeiger), Funktionen und Prozeduren und schließlich die Dateiverwaltung in Pascal.

(Anton Gruber/ Silvia Gutschmidt/cg)

Die Sache mit dem Semikolon:

Ein Semikolon trennt zwei Anweisungen voneinander. »Begin« ist keine Anweisung. Aber hinter dem »end« muß ein Semikolon stehen! Vor einem »end« kann die letzte Anweisung mit »;« abgeschlossen werden, muß jedoch nicht. Direkt vor dem »end« muß also kein Semikolon stehen. Vor einem »else« schließlich darf auf keinen Fall ein »;« stehen. Hier würde dieses Trennzeichen einen Syntaxfehler erzeugen.

Kennen Sie Ihren C 64? (Teil II)

Haben Sie sich eigentlich schon mal Gedanken darüber gemacht, wie der C 64 funktioniert? Wir wollen Ihnen in dieser Reihe zeigen, wie man dem C 64 auf die Schliche kommt.

Keine Angst — wir fangen langsam mit den Grundlagen an, um uns Stück für Stück vorzuarbeiten, bis der C 64 wie ein gläserner Computer vor uns steht. Was ist dran an der »Hintertür«, die wir im ersten Teil beschrieben haben? Was kann man damit machen? Nun, die Antwort ist relativ einfach, denn die »Hintertür« ist die einzige Möglichkeit, auf preisgünstige Weise ein beliebiges Standard-Diskettenlaufwerk mit FDC (Floppy-Disk-Controller) an den C 64 anzuschließen (Bild 1). Dazu wäre es interessant, zu wissen, wie dieser Eingang (SO) in der 6510-CPU tatsächlich funktioniert. Doch das ist ein Geheimnis der CPU-Entwickler. Besitzer der (längst vergriffenen) Original-Dokumentation über diese CPU finden nur den lapidaren Hinweis: Das ist zur Zusammenarbeit mit einem künftigen Baustein vorgesehen und sollte normalerweise nicht benutzt werden. Der »künftige« Baustein ist jedoch leider nie auf dem Markt erschienen.

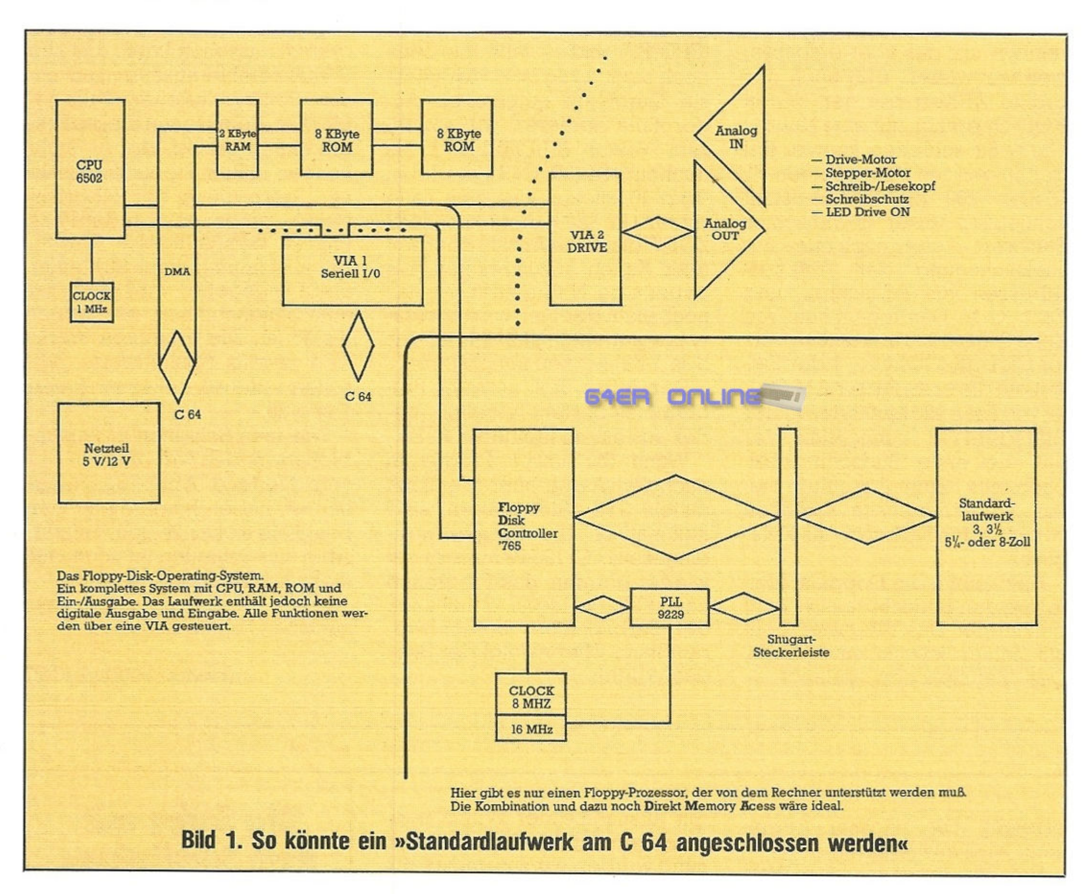


Bild 1. So könnte ein »Standardlaufwerk am C 64 angeschlossen werden«

Die Commodore-Entwickler haben beim C 64 diesen Signaleingang nicht mehr aus der 6510-CPU herausgeführt. Diese CPU ist übrigens nichts anderes als eine 6502-CPU mit einem Port und Steuereingängen für die Zusammenarbeit mit dem GDP (Grafik Display Prozessor oder Video-Chip). Nun gibt es eine unzählige Schar von Freaks und Hackern, die gerade beim C 64 auch den letzten POKE- oder PEEK-Befehl kennen. Doch wie sieht es mit der Sprache aus, die die 6510-CPU »spricht«? Assembler ist nicht nur Voraussetzung zum echten Messen, Steuern, Regeln und schnellen Spielen, sondern im Vergleich zum vorhandenen Basic (man könnte dem C 64 auch ein schnelleres Basic »einpflanzen«) fünf bis zehntausend mal schneller. Aufgrund jahrelanger Praxis in der Ausbildung kann behauptet werden, daß Assembler in der gleichen Zeit wie Basic erlernbar ist. Jede höhere Programmiersprache, und sei sie noch so komplex (zum Beispiel ADA oder C), muß am Ende von einer CPU ausgeführt werden. Und die »spricht« nur ihren speziellen Assembler.

Eine CPU kann man mit einem kompletten Computersystem vergleichen. Auch hier gibt es einen Interpreter (wie Basic ebenfalls interpretiert wird). Der Interpreter in der CPU bekommt über dem Datenbus ein Byte, und das wird in der CPU interpretiert. Das geschieht in dem Instruktionsregister des Prozessors. Ist das Byte nicht interpretierbar, »stürzt« der Computer ab und es hilft nur noch ein Reset.

Im Gegensatz zur Maschinensprache gibt es im Basic einen Notausgang. Wird ein Befehl nicht gefunden (weil er nicht in der Interpreterliste steht) oder kann er nicht ausgeführt werden (weil zum Beispiel keine Floppy angeschlossen ist oder per Befehl mathematisch nicht ausführbar ist, etc.), erzeugt das Betriebssystem (natürlich in Assembler geschrieben) eine Fehlermeldung. Die einfachste Fehlermeldung ist »SYNTAX ERROR«.

Wird ein Basic-Befehl eingegeben (oder im Programm abgearbeitet), sucht die CPU im Basic-Interpreter den Befehl und die entsprechende Routine dazu und führt sie aus. Die Routine ist natürlich wieder in Assembler geschrieben. Das alles braucht sehr viel Zeit.

Der eigentliche Chef

Doch das ist längst noch nicht alles. Nebenbei muß die CPU noch die Interrupt-Request-Routine (IRQ) bedienen. Sie wird ausgelöst durch den Timer eines Portbausteins (CIA 6526) des C 64. Die CPU unterbricht das Programm, rettet alle Register und führt die Routine aus. Diese Assembler-Routine wird 60mal pro Sekunde durchlaufen.

Dann ist da noch der Video-Chip VIC 6569 (GDP). Er sorgt für das Bild und die RAM-Auffrischung (Bild 2). Der GDP unterbricht dann die CPU, weil auf einem Bus ja nur einer arbeiten kann. Wenn der GDP auf dem Daten- und Adreßbus arbeitet, führt er direkte Speicherzugriffe (DMA) aus. Die Signalleitung dazu heißt »Bus available« (BA, Bus verfügbar) und geht zum CPU-Eingang »READY«. Das Signal

»BA« müßte eigentlich »DMA (Direct Memory Access) oder »Halt« heißen, denn wenn diese Signalleitung auf Low geht, wird der CPU sozusagen die »rote Karte« gezeigt. Das heißt, die CPU hat ihre Aktivitäten einzustellen, und von Daten- und Adreßbus zu verschwinden. Da sie das meistens nicht sofort kann, gibt ihr der GDP noch drei Taktzyklen Zeit, um »die Beine hochzunehmen«. Der Zugriff auf den Leitungen regelt die Signalleitung »Adress-Enable-Control« (AEC). Der Zustand High dieser Leitung bedeutet, daß die CPU aktiv ist, Low gibt dem GDP Vorrang. geht also »BA« auf Low, wartet der GDP noch drei Taktzyklen. Danach geht AEC so lange auf Low, bis die Arbeit des GDP beendet ist. Es ist sehr wichtig, daß vor dem Zugriff (DMA) noch drei Taktzyklen gewartet wird, da sonst das System abstürzt. Dieses ganze Timing muß auch ein Baustein »wissen«, der eigentlich die gesamte Koordinierung im System durchführt: das PLA-IC 251 064-01. Dieser Baustein ist ein programmierbares Logic-Array, das heißt Signale, die hineingehen, erscheinen an den Ausgängen je nach vorheriger Programmierung in Abhängigkeit der gewünschten Verknüpfung.

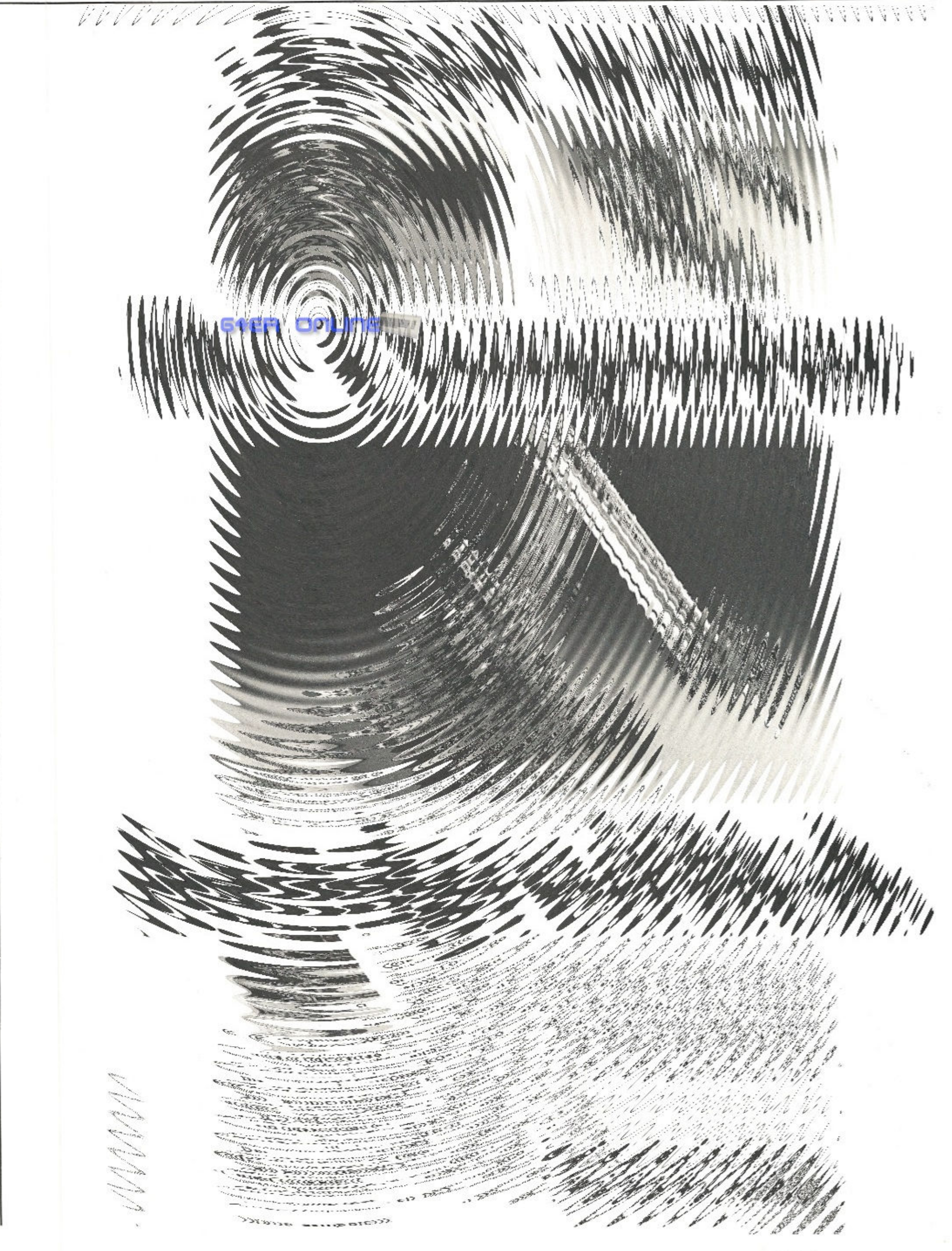
Dieser Adreßmanager ist für Entwickler ein alter Bekannter, nämlich das IC 82 S 100 von Intel oder Signetics. Commodore hatte bei den ersten Prototypen des C 64 auch dieses IC benutzt. Aber warum sollte es in Serie von Intel oder Signetics kommen, schließlich hat Commodore einen eigenen Prozessorhersteller im Haus: MOS Technologies.

Ob der GDP die CPU anhält, bestimmt wiederum am Anfang die CPU, denn beim Hochlaufen des Systems (Reset oder Kaltstart) wird der GDP auf Zugriff programmiert. Und wenn die CPU das macht, kann das ein Programmierer natürlich ebenfalls. Da alle diese Aktivitäten durch Steuerregister ausgelöst werden, kann man sowohl den GDP als auch den IRQ der CPU abschalten. Das in Basic geschriebene Beispiel (Bild 3 und Bild 4) zeigt eine beachtliche Zeitersparnis. Hier greift der GDP nicht mehr in den Lauf der CPU ein und auch die Interruptroutine braucht die CPU nicht mehr zu durchlaufen. Beim Sortieren von großen Feldern oder Rechnen mit komplexen Formeln empfiehlt sich diese Methode. Da der GDP abgeschaltet wurde, erscheint jedoch kein Bild. Auch verhindert die gesperrte IRQ-Routine eine Tastaturabfrage.

Alle Vorgänge werden im System durch Signale (Low und High) gesteuert. Der Systemtakt beim C 64 beträgt zirka 980 kHz, das heißt 980000mal pro Sekunde »passiert etwas« im System. Um verfolgen zu können, wer zu welchem Zeitpunkt was und wo macht, bleibt also wenig Zeit.

Nehmen wir einmal an, daß wir ein Programm im C 64 gestartet haben. Ist es ein Basic-Programm, so können wir im Problemfall ja mit einer Fehlermeldung rechnen, bei einem Assemblerprogramm bekanntlich nicht. Selbst die erfahrensten Assemblerprogrammierer verbringen Stunden mit einem Programmfehler, ohne ihn zu finden. Mit anderen Worten, wenn in diesem Artikel Schreibfehler sind, weiß der Leser meistens trotzdem, was gemeint ist — eine CPU aber verzeiht nicht den kleinsten Fehler. Fehlerbeseitigung in einem Programm ist also eine Frage des Wissens, der Zeit und der Meßmittel. Datenanalyser oder Datenlogger liegen leider preislich ab 20000 Mark aufwärts. Außerdem gibt es für den C 64 keine.

Diese Geräte zeichnen einen Signalverlauf im System auf. Der Adreßbus besteht zum Beispiel aus 16 Leitungen. Will die CPU ein Byte von einer Adresse holen, muß sie auf den 16 Leitungen die entsprechende Binärkombination setzen, um das Byte auf dem Datenbus »serviert« zu bekommen. Dazu muß noch der Baustein selektiert (Chip select) und das



Signal R/W gesetzt werden. Der C 64 arbeitet Low-Aktiv, das heißt wenn eine Leitung auf Low ist, ist das Bit gesetzt.

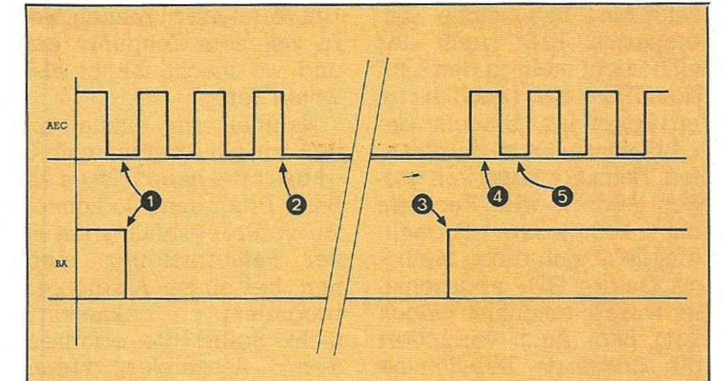
Um auf unser Problem zurückzukommen — wenn wir wissen wollen, was die CPU in Echtzeit bei welcher Adresse was macht, müssen wir diese Signale sichtbar machen. Man könnte ja auch die CPU nach jedem Schritt anhalten, um die Ausführung eines Programms zu überwachen, doch in Echtzeit (bei »echtem« Systemtakt) läuft es dann meistens doch nicht. Die Idee, durch Leuchtdioden die Signale sichtbar zu machen, scheint sehr abwegig. Das menschliche Auge hat ein Auflösungsvermögen von maximal 30 bis 40 Hz. Alle Leuchtdioden würden demnach auf einmal leuchten, und man würde keine Adresse oder Daten erkennen können.

Aber so unglaublich das klingt, genau diese menschliche Schwäche kann man sich beim Bau eines solchen Busmonitors zunutzemachen. Würde man diese Signale weiter ausdecodieren, und die Adresse auf einem Display dezimal oder hexadezimal anzeigen, könnte man in Echtzeit keine Adresse mehr erkennen. Doch zeigt man in Echtzeit den Wechsel zwischen Low und High mit einem Schaltungsstrick an, erkennt man klar, wo die CPU im System »herumläuft«.

Der Busmonitor arbeitet hier im Labor seit mehr als zwei Jahren problemlos. Er zeigt auf einen Blick, wo die CPU im Adreßraum »läuft« und welche Routinen sie benutzt. Außerdem kann er in erweiterter Form anzeigen, an welcher Adresse welches Byte verarbeitet wird.

Natürlich belegen diese Bus-Monitore weder Adreßraum noch sind sie auf Software angewiesen. Denn dadurch würden sie ja einen »normalen« C 64 verändern. Auch sogenannte Geheimbefehle im Assembler bereiten den Monitoren keine Probleme. Hier wird nur angezeigt, was eine CPU immer tun muß: Bei laufendem Systemtakt holt oder bringt die CPU ein Byte und dazu muß sie ja den Adreß- und Datenbus benutzen. Doch davon später mehr.

Der Monitor läßt sich auch gut benutzen zum Erlernen der Maschinensprache, da ein geschriebenes Programm jederzeit über den Monitor verfolgt werden kann. Wer einigermaßen mit dem Lötkolben umgehen kann und eine Investition von zirka 40 Mark nicht scheut, sollte seine Arbeitsmaterialien für unsere »Entdeckungsreisen« in den nächsten Folgen gleich bereitlegen. (Logo/aw)



Hier das Zeitdiagramm zur DMA. Die zeitliche Reihenfolge wird vom GDP gesteuert. Das Signal AEC vom GDP ist direkt mit den Adreßausgängen der CPU verbunden, das Signal BA mit dem Eingang Ready.
1 Wenn AEC Low ist und der GDP will DMA, geht das Signal »BA« auf Low.
2 Die CPU hat drei Taktzyklen Zeit gehabt, um ihren Systemzugriff abzuschließen. Jetzt bleibt AEC auf Low und schaltet den Adreßbus hochohmig, das heißt der Bus ist signalmäßig nicht vorhanden. Auch der Datenbus und das Schreib-/Lesesignal sind hochohmig geworden.
3 Hat der Master seine Arbeit beendet, setzt er das Signal »BA« auf High.
4 Kurze Zeit später geht das Signal AEC auf High und die CPU arbeitet wieder weiter im System.
5 AEC geht auf Low, doch die CPU arbeitet immer nur, wenn AEC High ist. Es wird also kein DMA durchgeführt.
DMA wird also nur von dem Signal »BA« ausgelöst. Dieses Signal steht auch am Expansion-Port zur Verfügung, da ein DMA von außen nur erfolgen darf, wenn der Master nicht im System ist.

Bild 2. Das Zeitdiagramm zum DMA (Direct Memory Access)

LINE	ADRESSE	CODE	MNEMO-MIC	TAKT-ZYKLEN
0001	1000	78	SEI	2
0002	1001	A9 00	LDA #$00	2
0003	1003	8D 11 D0	STA $D011	4
0004	1006	A2 00	LDX #$00	2
0005	1008	D0 F5	INC $1000	6
0006	100B	A0 64	LDY #$64	2
0007	100D	D0 FD	DEY	2
0008	100E	88	BNE $100D	2-3
0009	1010	E8	INX	2
0010	1011	EE 00 10	BNE $1008	2-3
0011	1013	A9 1B	LDA #$1B	2
0012	1015	8D 11 00	STA $D011	4
0013	1018	58	CLI	2
0014	1019	60	RTS	6

```
 1 REM DURCHLAUFZEIT MIT GDP UND IRQ ZIRKA 41,6 SEKUNDEN
 5 REM DURCHLAUFZEIT OHNE GDP UND IRQ ZIRKA 37,9 SEKUNDEN
 7 REM ZUM TESTEN ZEILE 11 ENTFERNEN
10 V=53265:C=56333:REM          ADRESSE FUER GDP EIN/AUS UND
                                CPU IRQ EIN/AUS
11 POKE V,0:POKE C,1:REM        GDP UND CPU IRQ AUS
20 W=100:M=4096:REM             ZAEHLER FUERS WARTEN UND SPEI-
                                CHERADRESSE
24 FOR T=0 TO 255:              SCHLEIFE MIT SPEICHER + 1
   POKEM,T:REM
26 FOR R=1 TO W:NEXT R:         WARTESCHLEIFE UND ABSCHLUSS
   NEXTT:REM
28 POKE V,27:POKE C,129:REM     GDP EIN UND IRQ EIN
   READY.
```

Bild 3. Mit diesen Programmen (Basic- und Assemblerversion) stoppen Sie den GDP

Beispiel in Basic

Erklärung des Programms (Bild 3) zum Abschalten des GDP und der IRQ-Routine der CPU.

Zeile 10: Die Registeradresse des GDPs und des Timerbausteins (VIA 6526) werden in die Variablen V und C übergeben.

Zeile 11: Der Zugriff des Masters im System wird gestoppt und die DMA-Zyklen werden nicht mehr ausgeführt. Der zweite POKE-Befehl stoppt die Timerausgabe. Dadurch erscheinen am Pin 3 der CPU keine IRQ-Impulse mehr und die CPU durchläuft nicht diese Betriebssystemroutine.

Zeile 20: Zur Verzögerung werden die Variablen für zwei FOR-NEXT-Schleifen initialisiert. W = Verzögerung kleine Schleife, M = Adresse zum Erhöhen.

Zeile 24: Große Schleife. In die Adresse M (4096) kommt der Wert (die Daten) T.

Zeile 26: Kleine Schleife mit der Verzögerung durch Variable W (100) und Abschluß der großen Schleife.

Zeile 28: Die Normalwerte werden wieder hergestellt.

Die Zeilen 20 bis 24 sind nur zur Laufzeitdemonstration. Werden die POKE-Befehle in Zeile 11 durchlaufen, wird die Tastatur nicht mehr abgefragt, die Uhr nicht nachgestellt und das Bild ist »nicht da«. Trotzdem sind bei zeitaufwendigen internen Rechneroperationen diese Routinen nicht von der Hand zu weisen, da eine beachtliche Zeitersparnis zu erreichen ist.

Beispiel in Assembler

Das zweite Programm ist in Maschinensprache geschrieben und bewirkt genau das gleiche.

Zeile 01: Set Interrupt disable Status (SEI). Der IRQ ist maskierbar, das heißt durch Setzen dieses Bits im Statusregister der CPU wird die Interrupt-Anfrage ignoriert (da kann der Takt von der VIA 6526 klopfen, wie er will). Basic-Programmierer müssen die Ursache (den Timer stoppen) abschalten — im Assembler reicht ein »Ein-Byte-Befehl«. Es gibt bei der CPU auch einen »Nicht Maskierbaren Interrupt« (NMI). Der wird durch die Tasten

Fortsetzung auf Seite 177


64ER ONLINE



Spielprogramme

Spiele programmieren muß nicht gleichbedeutend sein mit einer Menge Arbeit. Schlagender Beweis für diese Aussage ist der GameMaker von Activision. Außerdem können Sie mit dem GameMaker eine Reise nach San Francisco gewinnen.

Eine kleine Situationsbeschreibung: Sie haben sich gerade ein Spiel gekauft, packen es aus, laden es und fangen an zu spielen. Und nach einer Weile kommen Ihnen Gedanken ähnlich den folgenden: »Die Grafik ist an der Stelle aber sehr grob. Wer hat denn bloß diese bescheuerte Musik programmiert? Und die Punktewertung ist völlig unsinnig wie überhaupt das ganze Spiel. Ach, ein Programmiergenie müßte man sein, dann könnte man das alles sehr viel besser machen...«

Aber vor das eigene Spielprogramm haben die Götter nunmal den Programmierschweiß gesetzt — und der tropft heutzutage gleich literweise, da die Qualitätsansprüche gestiegen sind. Eigentlich muß ein anständiger Spieleprogrammierer auf den Gebieten Grafik, Sound, Programmierung und Kreativität ein Experte sein. Die großen Software-Firmen arbeiten deswegen nicht mehr mit Einzelprogrammierern, sondern mit kompletten Teams, in denen jeder seine Programmierspezialitäten ausspielen kann.

Was ist aber mit dem ambitionierten Computeranwender, der eine Idee für ein Spiel hat, sich aber aus verständlichen Gründen nicht an die Programmierung herantraut. Nun, dem Manne (und natürlich auch der Frau) kann geholfen werden. Ein neues Produkt von Activision namens »GameMaker« verspricht die Spieleprogrammierung wesentlich zu vereinfachen: »GameMaker ist sehr viel einfacher zu bedienen und doch sehr viel wirkungsvoller als normale Programmiersprachen« steht auf der Packungsrückseite.

»Einfacher« heißt aber nicht immer »einfach«. Denn auch der GameMaker kann aus einem Computer-Laien nicht so schnell einen Starprogrammierer machen. Als GameMaker-Benutzer müssen Sie nämlich immer noch programmieren. Dabei werden Sie allerdings so tatkräftig unterstützt, daß Sie sich auf die wesentliche Sache konzentrieren können: das Spiel.

Tolle Grafik mit wenig Aufwand

Die Problemkinder Grafik und Musik werden in mehreren, höchst komfortablen Editoren erstellt und dann mit einfachen Kommandos in das Spielprogramm eingebaut. Die vier Editoren heißen: »SceneMaker«, »SpriteMaker«, »SoundMaker« und »MusicMaker« — eine gewisse Linie bei der Namensgebung ist nicht zu übersehen.

Das wichtigste an einem Spiel ist meistens die Grafik, schafft sie doch einen ersten positiven oder negativen Eindruck beim Spieler. Die grafischen Möglichkeiten des GameMakers beschränken sich auf Hintergrundgrafiken und Sprites. Spiele mit geändertem Zeichensatz oder Scrolling sind also nicht möglich. Dadurch bleibt der GameMaker aber überschaubar und schnell in der Programmausführung.

Der SceneMaker (Bild 1), mit dem sich die Hintergrundgrafiken zeichnen lassen, ist ein einfaches Malprogramm, das alle wichtigen Funktionen enthält. Einen Nachteil muß man aber in Kauf nehmen: Es dürfen auf einem Bild immer nur vier verschiedene Farben auftauchen, die man aus der üblichen Palette von 16 Farben auswählen kann. Die Bedienung ist, wie auch in allen anderen Programmteilen, sehr komfortabel. Die Tastatur wird nur zum Eingeben des Filenamens beim Speichern eines Bildes verwendet. Alles andere wird durch den Joystick gesteuert. Etwas umständlich ist das Einfügen von Text ins Bild: Der SceneMaker enthält nämlich keine Text-Funktion. Das Handbuch schlägt unter der Rubrik »Tips & Tricks« folgende Verfahrensweise vor: Malen Sie das Bild ohne den Text, geben dann im Programmeditor mit Print-Befehlen den Text auf dem Bildschirm aus, gehen wieder zurück in den SceneMaker und speichern das nun fertige Bild.

Der fertiggestellte Hintergrund kann aus Geschwindigkeitsgründen während des Programmablaufs nur wenig geändert werden. Die GameMaker-Programmiersprache kennt deswegen nur einen Befehl zum Zeichnen von Punkten und Schreiben von Text. Objekte, die sich bewegen sollen, müs-

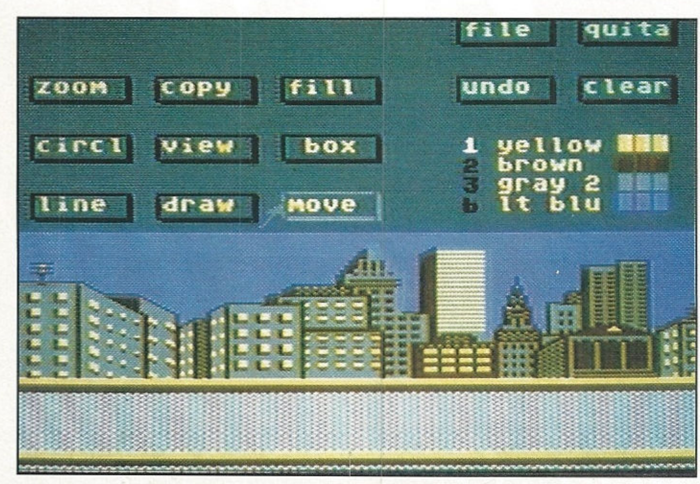


Bild 1. SceneMaker — Ein einfaches Malprogramm

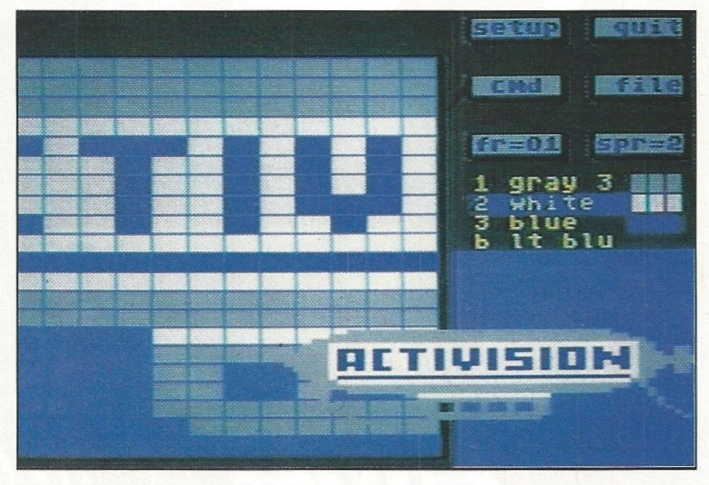


Bild 2. SpriteMaker — Animierender Sprite-Editor

im Selbstbau

sen mit Sprites dargestellt werden. Zur Definition der Sprites verwendet man den SpriteMaker (Bild 2).

Ein Objekt, sprich die Spielfigur oder andere »aktive« Teilnehmer am Spielgeschehen, braucht nicht notwendigerweise auf die Maße eines Sprites beschränkt sein. Der GameMaker erlaubt die Kombination von bis zu vier Sprites zu einem großen Objekt. Auch dies wird durch den SpriteMaker unterstüzt. Die vier Sprites können beliebig zusammengestellt werden. Allerdings sind nur vier verschiedene Farben (drei Farben und Transparenz) pro Objekt erlaubt. Nicht nur, daß man einzelne Objekte erstellen kann — Sie können diese sogar per Animation zum Bildschirmleben erwecken. Der GameMaker kann bis zu 31 Bewegungsstadien eines Objekts verarbeiten. Das drückt dann allerdings schon arg auf den zur Verfügung stehenden Speicherplatz. Irgendwelche Mängel oder Unzugänglichkeiten konnten wir im SpriteMaker nicht entdecken.

Der Ton macht die Musik

Zum guten Spiel gehört auch ein guter Ton. Egal ob Sound-Effekt oder Hintergrundmusik, vom GameMaker wird man auch schallmäßig komplett unterstüzt.
Der SoundMaker (Bild 3) ist ein kleines Synthesizer-Programm, das sich nicht zum Komponieren von Musikstücken, sondern nur von Sound-Effekten eignet. Ein Effekt kann aus verschiedenen Einzeltönen zusammengesetzt werden. Dabei ist die volle Kontrolle über den SID-Chip gegeben. Mit dem Joystick verstellt man Dreh- und Schieberegler, um von der Explosion bis zum Tarzan-Schrei die wildesten Geräusche zu erzeugen.

Anders aufgebaut ist der MusicMaker (Bild 4). Hier stehen nur 13 Klangfarben verschiedener Instrumente zur Verfügung, die dann in dreistimmigen Liedern eingesetzt werden können. Auch hier gilt: Fast alles wird mit dem Joystick gesteuert. Wie bei nahezu jedem Kompositionsprogramm üblich, werden Noten auf einem Notenblatt plaziert. Schade also, daß man von einem Programmierer musikalische Kenntnisse verlangt. Man hätte viel lieber nach einer Möglichkeit der Eingabe suchen sollen, die ohne Noten auskommt. Andererseits kann man so fertige Musik-Stücke »abtippen«.

Kommen wir aber zum Wichtigsten, der eigentlichen Programmierung des Spiels. Die GameMaker-Sprache ähnelt teilweise dem guten alten Basic. Allerdings fehlen einige Befehle, dafür sind sehr viele neue hinzugekommen.

Programmieren mit dem Joystick

Bei einem GameMaker-Listing (Bild 5) fällt als erstes auf, daß es keine Zeilennummern gibt. Um die vorhandenen Sprungbefehle (ähnlich GOTO und GOSUB) zu nutzen, muß man Labels vergeben. Ein Label besteht aus einer dreistelligen Nummer. Diese Nummer an sich ist nicht sehr aussagekräftig, so daß man als Programmierer ständig eine Liste führen sollte, welches Label auf welche Routine weist. Die Labels werden übrigens ausgerückt, das Listing wird dadurch und durch eine gute Farbwahl sehr übersichtlich. Zur Datenspeicherung benötigt man Variablen. Derer gibt es genau 26, von »a« bis »z« bezeichnet. Jede Variable kann einen Wert zwischen 0 und 255 annehmen. Außerdem gibt es zwei Score-Variablen, die Punktzahlen bis 999999 speichern können. Diese Score-Variablen werden vollautomatisch auf den Bildschirm abgebildet, wenn sich die Punktzahl ändert. Die Position dieser Punkteanzeige ist frei bestimmbar. Dies ist nur eines von vielen Programmierdetails, die die Arbeit mit GameMaker vereinfachen.

Für den fortgeschrittenen Programmierer gibt es noch weitere Datentypen, die allerdings schon komplizierter im Aufbau sind. So kann man eine RAM-Tabelle mit Konstanten anlegen, und, ähnlich wie in Basic, DATA-Zeilen benutzen.

Als besonders angenehm zeigt sich die Verwendung der in den vorherigen Programmteilen erstellten Grafiken, Sprites und Sounds. Ein Beispiel: Um ein Lied, das unter dem Namen »musik1« gespeichert wurde, automatisch zu spielen, gibt man einfach als Kommando »song is [musik1]« ein. Ist das Liedchen noch nicht im Speicher, wird es automatisch nachgeladen und gespielt. Ebenso verhält es sich mit Sprites, Sounds und Grafiken.

Wir haben leider nicht den Platz, um uns mit allen GameMaker-Kommandos zu beschäftigen. Eins ist aber sicher: Der Befehlssatz ist zwar einfach gehalten, bietet

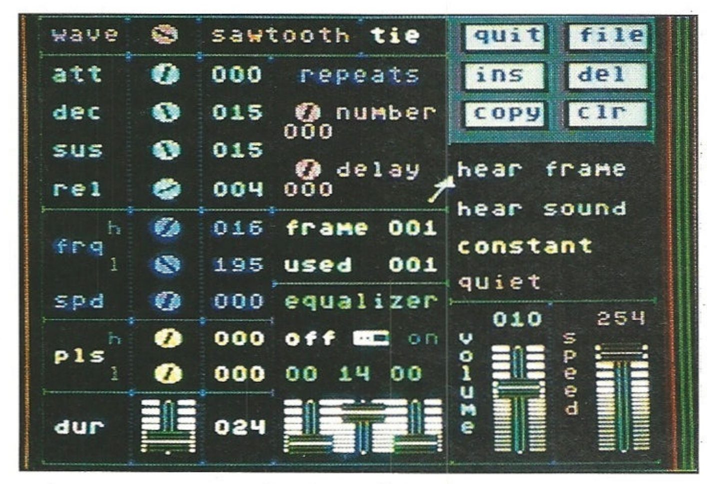


Bild 3. SoundMaker — Für jedes Geräusch zu haben

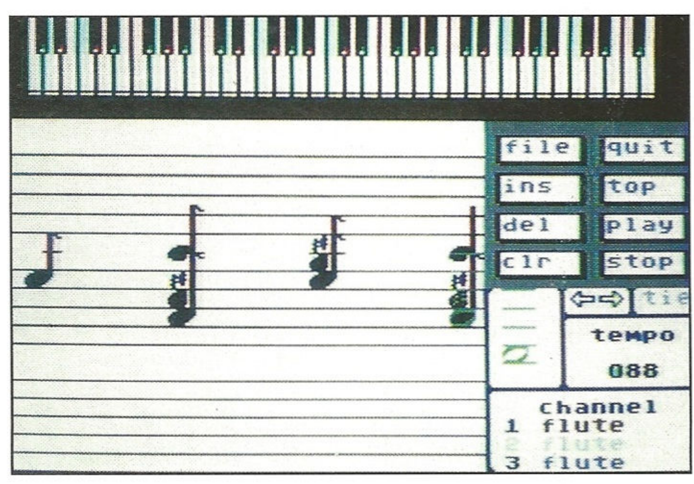


Bild 4. MusicMaker — Melodien flugs programmiert

aber alle notwendigen Befehle. Wer auf ein Problem stoßen sollte, welches er mit dem vorhanden Befehlssatz nicht lösen zu können glaubt, sollte einen Blick in die zahlreichen Beispiellistings werfen. Die Programmierer von Activision zeigen dort, was Sie alles aus dem GameMaker herausholen können, wenn man ein wenig trickst. Ein Beispiel ist das in Bild 6 gezeigte »Pitfall«. Dieser Videospiel-Klassiker enthält sehr viele Programmiertricks und sollte deswegen von jedem GameMaker-Besitzer durchgesehen werden.

Wer glaubt, daß es unmöglich sei, nur mit einem Joystick »bewaffnet« zu programmieren, wird vom GameMaker eines Besseren belehrt. Dafür ist etwas anderes unmöglich: die Tastatur zu benutzen. Man muß, und die Betonung liegt auf muß, die Befehle per Joystick aus einem Auswahlmenü entnehmen und im Listing ablegen. Das vermeidet zwar den typischen »Syntax (T)Error«, denn wer nicht tippt, kann auch keine Tippfehler machen. Wer allerdings schon öfters programmiert hat, wird sich nur schwer umstellen können.

Am Rande sei erwähnt, daß sich der GameMaker natürlich nicht nur für Spiele eignet. Grafik- und Musikdemos sind für den GameMaker nur »kleine Fische«. So lassen sich kleine Computerfilme oder auch elektronische Gratulationskarten entwerfen. Wer nicht gerade eine Tabellenkalkulation programmieren will, wird sicherlich noch viele andere Anwendungsmöglichkeiten für den GameMaker entdecken.

Keine Probleme mit dem Copyright

Das schönste am GameMaker kommt aber noch: Wählt man per Joystick die Option »Make-A-Disk« an, wird man aufgefordert, eine leere Diskette einzulegen. Dann wird das Spielprogramm gespeichert. Das Tolle daran ist, daß man das gespeicherte Spiel auch ohne den GameMaker spielen kann. Man kann die Diskette also einem Freund geben, ja darf und soll sie sogar kopieren! Das Copyright für das erstellte Spiel liegt alleine beim Programmierer. Er darf das Spiel beliebig weitergeben, ja sogar verkaufen. Voraussetzung ist nur, daß im Spiel ein Verweis auf den GameMaker erscheint. Dieser Verweis wird aber automatisch beim Speichern eingefügt. Damit werden zwei Fliegen mit einer Klappe geschlagen: Der Benutzer freut sich über diese faire Handlungsweise und der Hersteller hat so eine kostenlose Werbung. Um Mißverständnissen vorzubeugen: Diese Regelung gilt natürlich nicht für den GameMaker selber.

Über den Punkt Dokumentation kann man wieder nur zwiespältig berichten: Das Handbuch kann als gut bezeichnet werden, führt es doch anhand von Beispielen an den GameMaker heran. Eine mitgelieferte Kurzanleitung faßt auch alles wichtige auf wenigen Seiten zum Nachschlagen zusammen. Die gesamte Dokumentation ist aber in Englisch geschrieben, für den deutschen Markt wird nur die Kurzanleitung übersetzt. Durch das über 100 Seiten starke Buch muß man sich also mit seinem Schul-Englisch durchbeißen.

Zusammengefaßt kann man den GameMaker als sehr gut gelungenen Spielbaukasten bezeichnen. Obwohl er einfach zu bedienen ist, läßt sich sehr viel mit ihm anfangen. Schließlich muß man bedenken, daß man für den Preis von 89 Mark (Diskette) ein Grafik-, ein Sprite-, ein Musik- und ein Synthesizer-Programm zusammen mit einer schnellen Programmiersprache mit passendem Editor bekommt. Das Preis-/Leistungsverhältnis ist also mehr als ausgewogen.

Soll man die Grenzen des GameMaker definieren, fällt einem eigentlich nur eine ein: Er kann nicht auch noch die Spielidee liefern, denn Kreativität ist (noch) nicht programmierbar. (bs)

Info: GameMaker, Acitivision, Postfach 760680, 2000 Hamburg 76, zirka 89 Mark

64'er und Activision präsentieren den GameMaker-Programmierwettbewerb

GameMaker-Besitzer (und solche, die es werden wollen) aufgepaßt: Wir starten in Zusammenarbeit mit Activision einen GameMaker-Wettbewerb. Die Regeln sind recht einfach: Wer uns bis zum 30. Juni ein Spielprogramm, das mit dem GameMaker programmiert wurde, schickt, kann eine Reise nach San Francisco mit großem Empfang im Activision-Hauptquartier gewinnen.

Im Juli wird eine Jury aus Mitarbeitern der Markt & Technik Verlag Aktiengesellschaft und Activision das beste GameMaker-Spiel aus den Einsendungen heraussuchen. Für andere gute Spiele stellt Activision Software und Sachpreise zur Verfügung. Außerdem besteht die Möglichkeit, daß Activision Ihr Spiel oder Spielelemente kauft und weltweit vermarktet. Sie können ein professioneller Spieleprogrammierer werden!

Die genauen Teilnahme-Bedingungen stehen auf einer Karte, die jedem GameMaker-Original beiliegt. Über den Verlauf des Wettbewerbs werden wir ausführlich in der 64'er berichten. (bs)

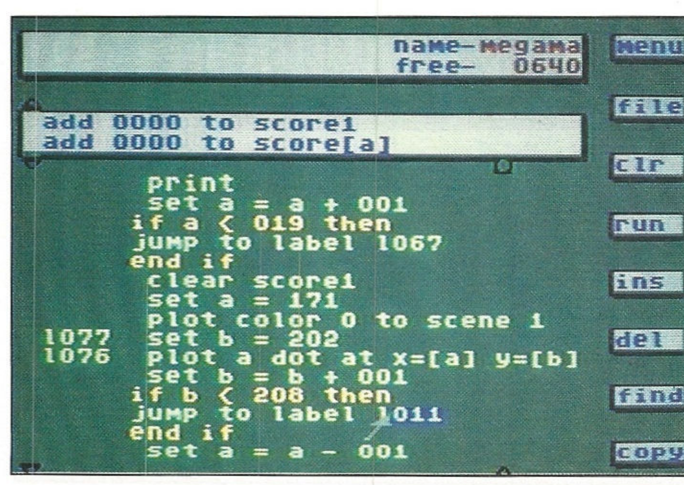


Bild 5. GameMaker — Programmieren mit dem Joystick

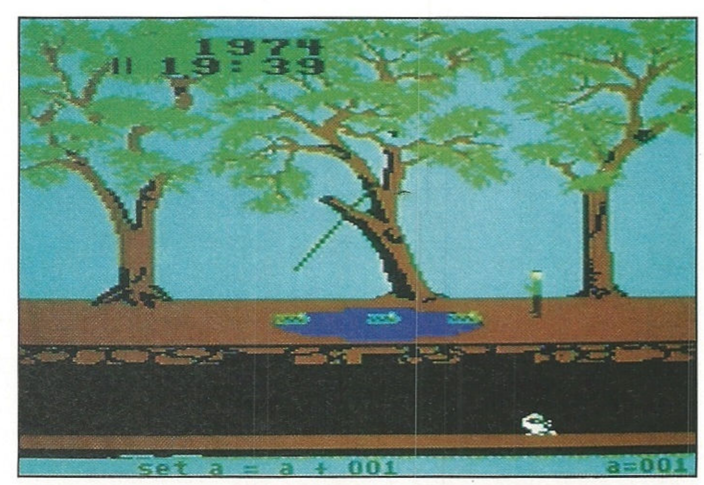


Bild 6. Pitfall — Ein typisches GameMaker-Spiel



Der zweite Preishammer

Ist das Dateiverwaltungsprogramm »StarDatei« dem »StarTexter« aus demselben Hause ebenbürtig? Sybex bietet mit der Datenverwaltung »StarDatei« ein weiteres Programm in der 64-Mark-Klasse.

Bekannt und berühmt ist das Textverarbeitungsprogramm »StarTexter«. Wer damit schon einmal gearbeitet hat, kommt auf Anhieb mit »StarDatei« gut zurecht. Viele Operationen sind ähnlich. Es gibt hier genauso einen Control-Modus, in dem man mit der »CTRL-Taste« gelangt (Bild 1), der auch die Menüauswahl (es stehen drei zur Verfügung) steuert.

Gleich nach dem Laden des Hauptprogrammes gelangt man in das Eingabefeld von »StarDatei«. Eine Maskendefinition, wie bei anderen Dateiverwaltungsprogrammen, wird hier nicht angewendet.

Jeder Datensatz hat immer ein Eingabefeld von 19 mal 40 Zeichen frei zur Verfügung. Um immer wiederkehrende Bezeichnungen, wie »Adresse:« oder ähnlichen nicht bei jedem Datensatz neu schreiben zu müssen, kann man sich eine komplette Seite »merken« lassen und bei Bedarf jederzeit auf den Bildschirm zurückholen. Es gibt keine festen Datenfelder mit einer erzwungenen Eingabe.

Menüwahl durch Control-Modus

Dieses System hat aber auch seine Nachteile. Es fehlen Hilfen wie TAP oder POS um schnell eine bestimmte Bildschirmposition zu erreichen. Ebenso wird bei jedem Datensatz die gesamte Information, also auch die Feldbezeichnungen wie »Name, Adresse oder Telefonnummer« mit gespeichert. Der Platzverbrauch auf der Diskette ist daher sehr groß.

Jeder Datensatz hat eine Titelzeile, die ein schnelles Auffinden ermöglicht. Nach ihr werden die Datensätze alphabetisch sortiert. Da die Daten in einem besonderen Algorithmus gespeichert

Suchen und Finden von Datensätzen

sind, wird die Diskette entsprechend formatiert. Jede Diskette darf nur eine Datei enthalten. Dieses ermöglicht dann in wenigen Sekunden ein Wiederauffinden einzelner Datensätze. Praktisch ist es, daß man jederzeit die Diskette und damit die Datei wechseln kann. Das Programm erkennt einen Diskettenwechsel automatisch und reagiert dann entsprechend.



Bild 1. Das Hauptmenü von StarDatei 64

Bild 2. Ein Datensatz auf dem Bildschirm

Einen guten Eindruck macht »StarDatei« beim Suchen und Finden von Datensätzen. Mit den von der Floppy bekannten Jokerzeichen »*« und »?« lassen sich die Kopfzeilen sekundenschnell durchsuchen und auf dem Bildschirm anzeigen. Mehrere Datensätze mit gleichem Suchwort werden alphabetisch aufgelistet.

Der Joker erleichtert die Arbeit

Damit ist die Leistungsfähigkeit keineswegs ausgeschöpft. Man kann auch verschiedene Suchbegriffe miteinander verknüpfen und so zum Beispiel bei einer Adreßdatei alle Leute, die in München wohnen und Meyer heißen, raussuchen. Man nennt dieses auch eine UND-Verknüpfung. Aber auch ODER-Verknüpfungen sind möglich. Beispielsweise lassen sich so alle Leute, die in München oder in Augsburg wohnen, bei einer angelegten Adreßdatei herausfinden. Die UND- beziehungsweise ODER-Verknüpfung kann im Suchbegriff auch mehrmals vorkommen, aber leider nicht gemischt.

Es können natürlich nicht nur die Titelzeilen durchsucht werden, sondern auch einzelne Teile aus den Datensätzen selber. Das dauert aber länger, da dazu auf die Diskette zugegriffen werden muß. Jeder gefundene Datensatz wird dann wieder in alphabetischer Reihenfolge auf dem Bildschirm aufgelistet und man kann auswählen, ob man ihn in den Speicher übernehmen oder weiterblättern will. Dabei lassen sich auch einzelne Datensätze markieren, beispielsweise um sie dann später auszudrucken oder mit »StarTexter« weiter verarbeiten zu lassen (Bild 2).

Bemerkenswert gut gelöst sind die diversen Möglichkeiten, sich einen Überblick der vorhandenen Daten zu verschaffen. Läßt man jeden Datensatz durch ein Zeichen auf dem Bildschirm anzeigen, erhält man eine gute Übersicht der Datenmengen. Fährt man jetzt mit dem

Immer die Übersicht!

Cursor durch diese Anzeige, bekommt man die Titelzeile des jeweiligen Datensatzes gezeigt, der sich auf Wunsch vollständig auf dem Bildschirm anzeigen läßt. Auf Wunsch kann man sich einzelne Datensätze markieren, wobei sich das verwendete Jokerzeichen in ein volles Quadrat ändert. Diese markierten Datensätze lassen sich in einer Liste zusammenfassen und nacheinander ausdrucken.

Die Druckerunterstützung

Die möglichen Fähigkeiten, die die einzelnen Drucker anbieten, werden bei weitem nicht voll ausgeschöpft. Das sind zum Beispiel Unterstreichen, Fettschrift oder auch das Format des Ausdruckes, das sich nicht frei wählen läßt. Es wird immer der Bildschirminhalt vom Drucker ausgegeben. Sinnvoll dagegen ist die Möglichkeit, sich jederzeit eine Hardcopy des Bildschirms ausgeben lassen zu können.

Nachteilig ist, daß aus dem Inhaltsverzeichnis einer Datendiskette sich nicht erkennen läßt, wieviel Speicherkapazität noch zur Verfügung steht. Man muß sich bei Bedarf einen Statusreport geben lassen, der den prozentualen Speicherverbrauch angibt. Ferner wird die Anzahl der im Moment gespeicherten Datensätze angezeigt.

Datenex- und Datenimport

Wichtig bei Dateiverwaltungsprogrammen ist vor allem der Datenexport, also die Weitergabe gesammelter Daten an andere Programme, zum Beispiel Textverarbeitungen. Leider arbeitet »StarDatei« nur mit dem »StarTexter« aus gleichem Hause zusammen. Der »StarTexter« kann die in dem speziellen Format abgelegten Daten lesen und in seinen Textspeicher übernehmen, oder auch beim Ausdrucken von Formbriefen verwenden.

Gut gelöst ist die Möglichkeit, einzelne Datenfelder mittels spezieller Zeichen für »StarTexter« zu definieren. Die so festgelegten Datenfelder lassen sich in beliebiger Reihenfolge von »StarTexter« weiterverarbeiten, so daß über die Textverarbeitung Listendruck von Datensätzen oder diverse Serienbriefe nach unterschiedlichsten Verarbeitungskriterien möglich ist. Ein weiterer Datenexport aus der »StarDatei« auf andere Textverarbeitungsprogramme ist leider nicht möglich. Dieser Umstand muß als echtes Manko bezeichnet werden. Die Übernahme von Daten durch »StarDatei« in seinen Speicher ist nur eingeschränkt möglich. Man kann eine SEQ-Datei in den Arbeitsspeicher einlesen und auf dem Bildschirm anzeigen lassen. Diese Daten lassen sich dann als Datensatz speichern, wenn sie nicht mehr als drei Blöcke auf der Diskette belegen.

Wie wäre es mit »Basic«?

Man kann jederzeit eine mathematische Berechnung durchführen. Dazu gibt man die Berechnung ins Datenfeld ein und drückt CTRL und »=«. Sofort bekommt man das Ergebnis in der nächsten Zeile darunter angezeigt. Es kann im Datensatz weiterverarbeitet werden. Selbst kleine Basic-Anweisungen innerhalb der »StarDatei« sind kein Problem. Wem das allerdings immer noch zu wenig ist, der kann für kurze Zeit »StarDatei« ganz verlassen und ein wenig in Basic programmieren. Man muß allerdings darauf achten, daß man nach jeder Kommandozeile ein »:STOP« anfügt, ansonsten wird »StarDatei« sofort wieder aktiviert.

Die oberste Bildschirmzeile ist die Kommunikationszeile, wenn man Anweisungen im Text übernehmen möchte. Eine hier plazierte Berechnung wird von »StarDatei« übernommen.

Die Parameteranpassungen

Jeder Ausdruck auf einem Drucker setzt die richtige Anpassung an dessen Fähigkeiten und Eigenheiten voraus. Da »StarDatei«-Druckmöglichkeiten wie Fettdruck oder Unterstreichung nicht unterstützt, fällt diese Druckeranpassung mager aus. Das Programm fragt dabei im Menü »Parameter« nach den wichtigsten Druckereinstellungen wie Wandlungsart, automatischer Zeilenvorschub, Anzahl der Spalten zum Einrücken beim Ausdruck und noch einiges mehr. Man kann hier wahlweise Drucker über den seriellen Bus oder über eine am User-Port simulierte Centronics-Schnittstelle ansprechen. Die Druckcodes von Umlauten, die man im Text frei verwenden kann und die normalerweise auf den Funktionstasten liegen, lassen sich hier ebenfalls festlegen. Des weiteren ist der von jeder Textverarbeitung her bekannte Zeilenumbruch, der beim Schreiben ein zu lang geratenes Wort vollständig in die nächste Zeile zieht, möglich. Diesen Zeilenumbruch kann man wahlweise ein- oder ausschalten.

Fast schon wie eine Schreibmaschine

Normalerweise entspricht die Tastaturbelegung des C 64 nicht der deutschen Schreibmaschinennorm nach (QWERTZ). Um hier jedoch trotzdem alle Umlaute eingeben zu können, wurden sie auf die Funktionstasten gelegt. Wem das zu umständlich ist, kann auch auf eine deutsche Tastenbelegung umschalten, wobei jetzt alle Zeichen dort liegen, wo sie auch bei einer deutschen Schreibmaschine zu finden sind. Auch »z« und »y« sind vertauscht.

Selbstverständlich lassen sich die Bildschirmfarben an individuelle Neigungen anpassen. Interessant, und auch von »StarTexter« her bekannt, ist die Möglichkeit, einen beliebigen Zeichensatz auszuwählen, der dann beim nächsten Neustart des Programmes mit eingeladen wird. Diese ganzen Parametereinstellungen lassen sich speichern und werden automatisch wieder aktiviert, wenn das Programm neu eingeladen und gestartet wird.

Fazit

Für einfachere Datenverwaltung, insbesondere das Führen von Adreßdateien oder auch beispielsweise Plattensammlungen, läßt sich »StarDatei« sehr gut verwenden. Wer zudem schon das Textverarbeitungsprogramm »StarTexter« besitzt, kann die gesammelten Daten sehr gut übernehmen und mannigfaltig weiterverarbeiten. Dazu kommt aber das Problem, Daten nicht frei an andere Textverarbeitungsprogramme weitergeben zu können. Auf der anderen Seite überzeugen jedoch Leistungen wie das einfache Eingeben neuer Datensätze ohne umständliche Maskendefinition mit endgültiger Festlegung aller Datenfelder. Man bekommt jedesmal einen Textbereich zur freien Verfügung gestellt und kann jederzeit die Organisation eines Datensatzes umstellen, sogar eine Notizzettelsammlung mit ungeordneten Datensätzen aufstellen.

Ferner bemerkenswert ist die Möglichkeit, kurzzeitig das Programm zu verlassen und kompliziertere, mathematische Berechnungen schnell in Basic durchzuführen. Bei Betrachtung all dieser Fähigkeiten bekommt man für 64 Mark ein gutes Programm. Wie bei Redaktionsschluß bekannt wurde, folgen weitere Kompatible Programme zum StarTexter und zur StarDatei. So zum Beispiel der StarPainter, ein Malprogramm.

(Karl Hinsch/do)

Info: DTM, Data Technology Management, Borahoferweg 5, 6200 Wiesbaden, Tel.: (06121) 407989, Preis 64 Mark.



Bild 1. »Revs« und »Jet«, die beiden Super-Simulationen

Die Super-Simulationen

In Fortsetzung unseres ersten Simulations-Artikels in Ausgabe 1/86 stellen wir Ihnen diesmal zwei neue Simulationsprogramme vor: Revs und Jet.

Simulationen mit Action-Elementen gewinnen immer mehr an Beliebtheit. Zwei neue Programme fahren, beziehungsweise fliegen, voll auf dieser Welle mit: »Revs« ist eine Formel-3-Simulation erster Güte, während man mit dem »Jet« in die Lüfte geht.

Ob in der Luft...

SubLogic, die Produzenten des Verkaufshits »Flight Simulator II«, versuchen mit »Jet« eine mehr kriegerisch gehaltene Simulation unter die Leute zu bringen.

Beim Auspacken des »Jet« findet man einige Beigaben, allerdings bei weitem nicht so viele wie beim »Flight II«. Ein dünnes DIN-A3-Handbuch und eine Pappkarte müssen genügen. Die Karte enthält in Kurzform alles, was man über den Jet wissen muß, das Handbuch sehr detaillierte Informationen über Programm und Flugzeuge.

Simuliert wird nicht nur einer, sondern gleich vier Jets. Der Spieler kann zwischen einer landgestützten F-16 und der Flugzeugträger-Version, einer F-18, wählen. Startpunkt für den Ausflug ist dann jeweils ein ordinärer Hangar oder der Flugzeugträger »U.S.S. Nimitz«. Und damit die Fliegerei nicht langweilig wird, simuliert der Computer noch ein paar russische MiG-21 und MiG-23 mit, über die der Spieler verständlicherweise keine Kontrolle hat.

Die gespeicherte Landschaft des Jet ist etwas dürftig ausgefallen. Man muß sich mit großen Land- und Wasserflächen, garniert von einigen feindlichen Stützpunkten, begnügen. Städte und geographische interessante Landschaften sind nicht zu finden. Dafür darf man Disketten des »Flight II« im Jet als Landschaftsdisketten verwenden.

Der Spieler hat sich durch mehrere Auswahlmenüs zu »kämpfen« bevor er abheben darf. So wird er als erstes gefragt, ob er's lieber bunt oder schwarzweiß hätte. Das zweite Menü ist das wichtigste, stellt man doch hier den Spiel-Modus ein. Der Modi gibt es vier, die da lauten:

»Dogfight«: Aufgabe ist es, möglichst viele russische Flieger vom Himmel zu holen, bevor man selber abgeschossen wird. Die Russen schießen nämlich äußerst scharf zurück. Wer dreimaligen Abschuß durch rechtzeitiges Betätigen des Schleudersitzes überlebt, wird vom Dienst suspendiert, oder anders ausgedrückt: »Game Over«.

»Target Strike«: Hier sind russische Land- und Seeziele (je nach Standort) zu vernichten. Die Gegner wehren sich diesmal mit Boden/Luft-Raketen. Auch hier gilt: Drei Flugzeuge darf man verlieren, dann ist Schluß.

Für etwas friedfertigere Gemüter gibt es den »Free Flight«. Hier darf man seinen Jet mal ausfliegen, ohne gleich schießen zu müssen. Dieser Modus ist ideal sowohl zum Üben wie auch für Kunstflug.

Der vierte Modus ist recht uninteressant, denn hier übernimmt der Computer alle Rollen und führt so ein zünftiges Demo eines »Dogfight« vor.

In weiteren Menüs ist der Schwierigkeitsgrad, der Flugzeugtyp und die Bewaffnung zu wählen. Der Spieler kann AIM-9-Side-Winder- und AIM-7-Sparrow-Raketen, ein M-61-Maschinengewehr sowie AGM-65-Maverick- und MK-82-Smart-Bomben mit an Bord nehmen. Was und wieviel man mitnehmen sollte, hängt vom Modus und Schwierigkeitsgrad ab.

Nun geht es endlich los. Sie haben sich eine F-18 gewählt und stehen auf dem Startkatapult des Flugzeugträgers. Sie schalten ihre Nachbrenner ein und in dem Augenblick, indem Sie vollen Schub erreichen, lösen Sie das Katapult aus. Mit irrsinniger Beschleunigung fegen Sie über das Deck und rasen kurz darauf mit doppelter Schallgeschwindigkeit über das Wasser. In Ihren Ohren klingt das geschickt programmierte Heulen der Düsenmotoren. Der Bildschirm zeigt in 3D-Grafik Wasser, Land, Gebäude, Flugzeuge und Schiffe.

Flight-Simulator-II-Fans, die eine stark verbesserte Grafik erwartet haben, werden enttäuscht. Die Grafik des Jet ist nur unwesentlich schneller. Dafür wurden mehr Zusatzfunktionen in den Jet gepackt. So kann man, ähnlich einem Fernglas, weit entfernte Objekte heranzoomen. Wer seinen Jet mal von außen sehen will, kann sich per Knopfdruck in den Kontrollturm versetzen lassen. Sehr witzig ist diese Option, wenn man bei gelandetem Jet den Schleudersitz betätigt: Man sieht deutlich, wie ein kleines Strichmännchen in die Luft geschleudert wird, um dann sanft an einem Fallschirm zu Boden zu schweben.

Wem die andauernde Schießerei nichts ausmacht, bekommt mit Jet einen hervorragenden Simulator. Hier wurde fast alles berücksichtigt, egal ob Aerodynamik der Flugzeuge und Raketen oder Einwirkung der Fliehkraft auf den Menschen. Ist diese zu hoch oder zu niedrig, führt dies zu den gefürchteten Red- und Blackouts: Der Pilot verliert das Sehvermögen, kurz darauf das Bewußtsein und natürlich die Kontrolle über das Flugzeug. Zusammengefaßt ist Jet eine Flug-Simulation erster Klasse, die im Punkt Realität höchstens vom »Flight Simulator II« überflügelt wird.

...oder zu Land

Wer sich nicht in die Lüfte begeben, sondern auf festem Boden bleiben will, könnte mit »Revs« seine Simulation gefunden haben.

Autorennspiele gibt es viele, echte Autorennsimulationen sind selten. Doch mit »Revs« gibt es nun ein Programm, das den Schwerpunkt voll auf die Simulation legt, ohne daß der Spielspaß zu kurz kommt. Gefahren wird mit einem RALT-RT3-Toyota auf den britischen Weltmeisterschaftsstrecken Silverstone und Brands Hatch. Über Auto wie Strecke geben zwei Anleitungshefte detaillierte Informationen. Wichtigster Unterschied zwischen einem Formel-3- und dem heimischen Volkswagen (oder anderem Serienauto): Einen Formel-3 fährt man nicht nach Tachometer, sondern nach Drehzahlmesser, im Fahrerjargon liebevoll »Revs« genannt. Bei einem Rennen kommt es stets darauf an, den Motor optimal auszunutzen, die hohe Geschwindigkeit ergibt sich dann von selbst. Wird

eine Höchstdrehzahl überschritten, kann man ernsthafte Probleme mit dem Motor bekommen.

Bevor man allerdings die Strecke seiner Wahl befährt, müssen die Spoiler des Autos eingestellt werden. Hier sollte ein möglichst guter Kompromiß zwischen Bodenhaftung und Luftwiderstand geschlossen werden. Dieser Thematik ist übrigens ein ganzes Kapitel in einem der Hefte gewidmet.

Einer gegen neunzehn

Wer will, kann jetzt gleich gegen 19 Mitkonkurrenten fahren, oder erst einmal üben, was sich gerade für Anfänger empfiehlt. Danach heißt es Zündschlüssel drehen, Gas geben, Gang einlegen und los geht's. Der Wagen legt viel Wert auf korrektes Schalten, sonst liegt man entweder im Graben oder steht bewegungslos auf der Strecke. Ist man schnell genug, geht es ans Überholen der Gegner. Das ist nicht ganz einfach, da manche Fahrer einen durch riskante Manöver abdrängen wollen. Große Hilfe bietet hier der Rückspiegel, der herannahende Autos klar zeigt. Gesteuert wird mit Tastatur, Joystick oder Paddles. Einige wichtige Funktionen werden immer über die alphanumerische Tastatur kontrolliert.

Die beiden Rennstrecken wurden ihren britischen Originalen sehr genau nachempfunden, es wurden sogar deren Höhenunterschiede berücksichtigt. Der Wagen verhält sich bei Steigungen anders, als wenn es bergab geht; und wer zu schnell fährt, wird bei kleineren Hügeln kräftig durchgeschüttelt.

Im Rennen fährt man gegen 19 Mitstreiter, deren Namen auf einen Sinn für Humor beim Programmierer schließen lassen: Miles Behind, Davey Rocket oder Peter Out geben sich hier die Ehre. Wer es auf ein Turnier im Kreise der Freunde oder Familie anlegt, wird ebenfalls bedient. In einer Vorrunde darf jeder Spieler eine Runde über den Kurs drehen. Dabei wird einerseits seine Startposition bestimmt. Andererseits merkt sich der Computer den Fahrstil jedes einzelnen Fahrers. Im eigentlichen Rennen, das jeder Spieler alleine fahren muß, werden die menschlichen Gegner dann einfach mitsimuliert! So kann man gegen seine Freunde fahren, obwohl nur einer am Computer-Steuer sitzen kann.

Für die Formel-3-Neulinge gibt der weltbekannte Fahrer David Hunt in einem der beiden Hefte viele Tips. Er beschreibt die Strecke, mögliche Überholmanöver, die Eigenschaften der Computergegner und vieles andere mehr. Ein Poster der Rennstrecken rundet den Pakkungsinhalt ab.

Mit Revs liegt eine ausgezeichnete Fahrsimulation vor, die realistisch und fesselnd zugleich ist. Mit dem hohen Realitätsgehalt ist aber auch ein hohes Frustrationselement hinzugekommen, denn Revs ist wahrlich kein leichtes Spiel. Wer eine Runde heil besteht, und unter 3 Minuten bleibt, darf sich schon als Halb-Profi sehen.

Damit ist Revs die beste Auto-Simulation, die es derzeit gibt. Man darf sie ruhig mit dem Flaggschiff der Flugsimulationen, »Flight II«, vergleichen.

1986 scheint also ein gutes Jahr für die Simulations-Liebhaber zu werden. Ein grandioser Anfang ist auf jeden Fall gemacht.

(Manfred Kohlen/bs)

Info: Jet, Softline, Schwarzwaldstr. 8a, 7802 Oberkirch, ca. 160 Mark (Diskette)
Revs, Rushware, An der Gümpgesbrücke 24, 4044 Kaarst 2, ca. 80 Mark (Diskette)

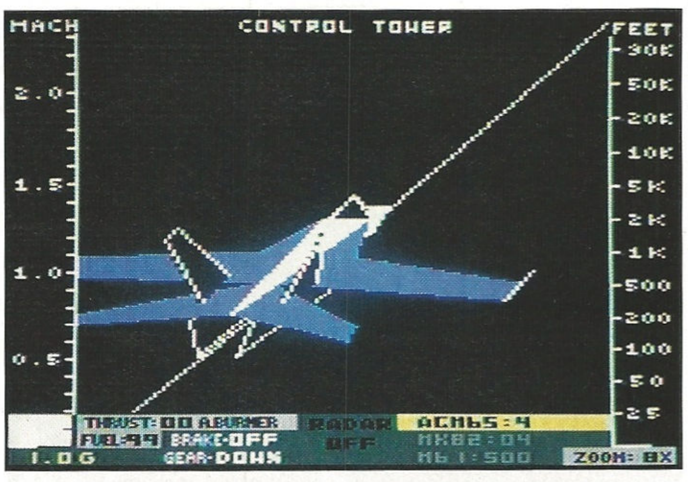


Bild 2. Der Jet, eine F-18, beim Start vom Flugzeugträger



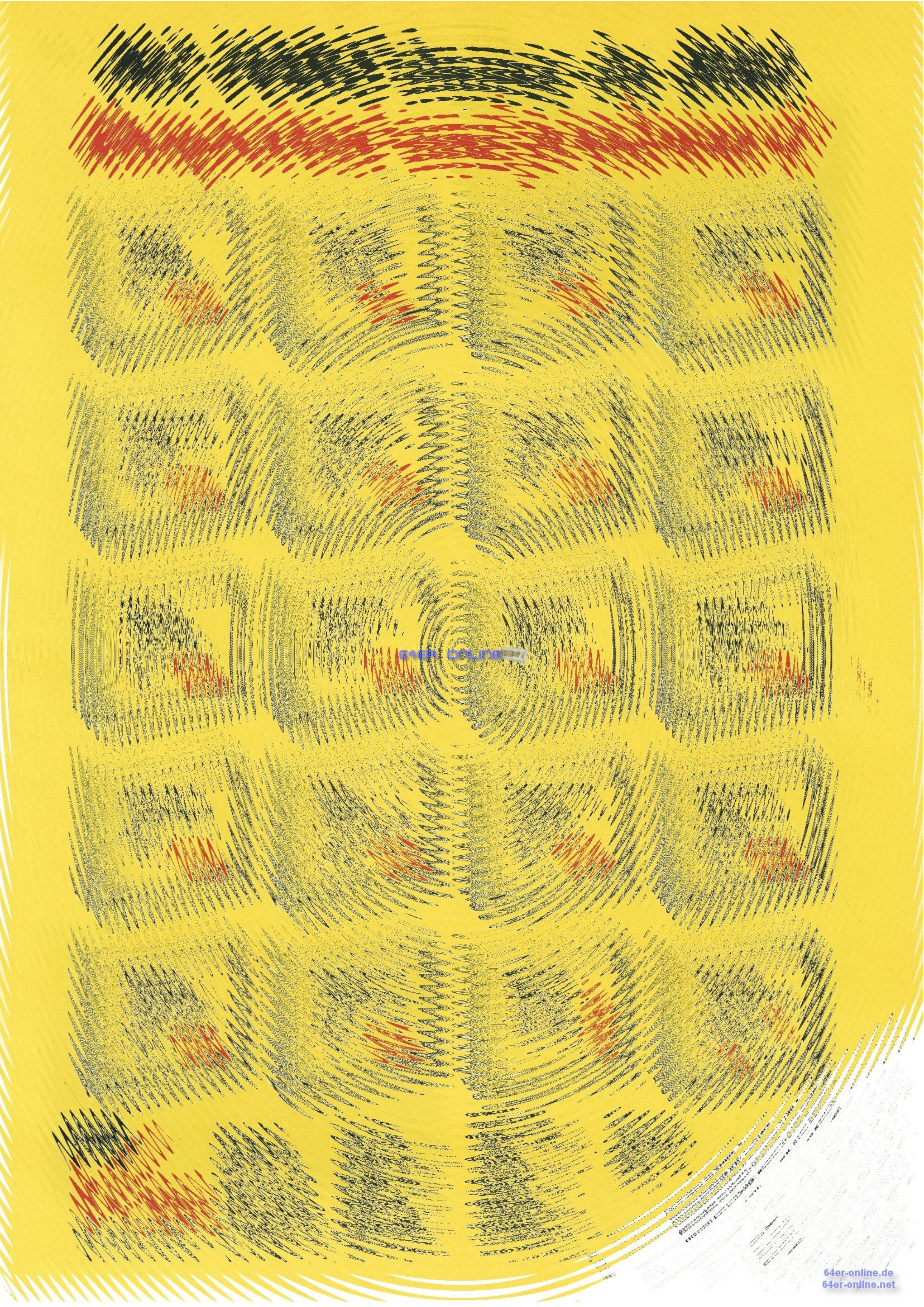
Bild 3. Eine typische Kampfszene, live aus dem Jet



Bild 4. Spannender als echte Formel-3: Revs

Kurz getestet: Scenery Disks

Nach monatelanger Wartezeit sind jetzt endlich die ersten Zusatzdisketten für den »Flight Simulator II« lieferbar. Von diesen sogenannten Scenery-Disks gibt es inzwischen sechs Stück, die den gesamten Osten der USA abdecken. Wer nun aber frohen Mutes in den Laden stürzt, um die Diskette Nummer CM-SD3 zu bestellen und dann über San Francisco seine Kapriolen zu ziehen, den müssen wir warnen: Auf den Disketten befinden sich nur die wichtigsten Landschaftsdaten wie Flüsse, Autobahnen und Flughäfen. Andere, optisch ansprechendere Gegenstände wie Städte und Gebäude fehlen fast völlig. Durch diese Beschränkung ist der Wert dieser Zusatzdisketten stark gemindert. Dafür sollen demnächst sogenannte Star-Disketten einzelner Städte erscheinen, auf denen wirklich alle wichtigen Gebäude, Sehenswürdigkeiten und ähnliches gespeichert sein sollen. (bs)





The Last V8

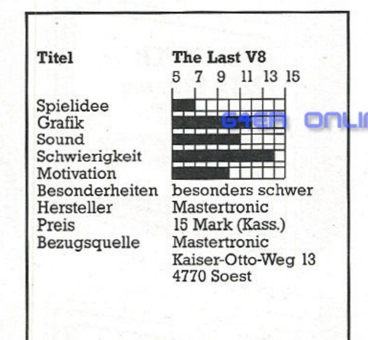
Die verrückten Billigspiele

M.A.D.-Games, übersetzt verrückte Spiele, ist eine neue Reihe von Programmen, die für 15 Mark eine ganze Menge bieten. Wir hatten jedenfalls viel Spaß mit den neuen Billigspielen.

Wer den Namen Mastertronic hört, denkt unweigerlich an »Billigspiele«, oder anders ausgedrückt, an Spiele mit taschengeldgerechten Preisen. Daß die Qualität bei einem Preis von 10 Mark nicht leiden muß, hatte diese Firma schon öfters bewiesen, obwohl es auch Ausreißer in negativer Richtung gab. Doch mit einer neuen Produktserie versucht Mastertronic, vom Billigspiel-Image wegzukommen.

Es handelt sich um die »M.A.D.-Games«, eine Reihe von bisher drei Spielen. (»M.A.D.« ist eine Abkürzung für »Mastertronic Added Dimension«). Sie kosten auf Kassette jeweils 15 Mark, und lassen sich von Grafik, Musik oder Spielwitz mit anderen Neuerscheinungen, die 30, 40 oder 50 Mark kosten, vergleichen. Die drei Titel lauten »Hero Of The Golden Talisman«, »The Last V8« und »Master Of Magic«.

Viel interessanter sind die beiden anderen »M.A.D.«-Spiele. Mit »The Last V8« erhält der Käufer ein Programm mit sehr guter Grafik und ebenso guter Musik. Das Spielprinzip ist einfach, die Hintergrundgeschichte kompliziert: Ein Auto muß verschiedene Strecken während bestimmter Zeitlimits zurücklegen. Dieses Auto, der letzte V8, ist das letzte, vom Atomkrieg verschonte Gefährt. Der Spieler muß es durch radioaktiv verseuchte Städte zu Basen anderer Überlebender lenken. Einige verspätete Atombomben erschweren diese Aufgabe.



Titel	The Last V8
	5 7 9 11 13 15
Spielidee	
Grafik	
Sound	
Schwierigkeit	
Motivation	
Besonderheiten	besonders schwer
Hersteller	Mastertronic
Preis	15 Mark (Kass.)
Bezugsquelle	Mastertronic Kaiser-Otto-Weg 13 4770 Soest

Viel Arbeit wurde in Grafik- und Sound-Effekte gesteckt. Hintergrundgrafik und Musikthema können locker mit fast jedem viel teureren Spiel konkurrieren. Lediglich die digitalisierte Sprache am Anfang einer Runde hätte man sich sparen können.

Der einzige Haken an »The Last V8« ist die Schwierigkeit. Die abzufahrenden Strecken sind sehr kurvenreich und eng, die Zeitlimits extrem knapp gewählt und die Joysticksteuerung stark gewöhnungsbedürftig. Zusammengefaßt kann man das Spiel als fast unmöglich bezeichnen — uns ist es jedenfalls noch nicht gelungen, die erste Basis zu erreichen. Aber gerade dieser Punkt dürfte »The Last V8« für diejenigen interessant machen, die eine echte Herausforderung suchen.

Auch das nächste »M.A.D.«-Spiel hat uns angenehm überrascht. »Master Of Magic« ist ein nicht allzu komplexes Rollenspiel mit Elementen, die aus Adventures und Actionspielen stammen.

Der Spieler wird, als er eine Höhle erforscht, von einem bösen Zauberer entführt. Dieser will den Armen erst wieder freilassen, wenn er ihm ein bestimmtes Amulett besorgt hat. Die Amulettsuche ist natürlich kein leichtes Unterfangen, da sich zahlreiche böse Monster in den finsteren Höhlen-Gängen herumtreiben. Dort verstreute Waffen und Gegenstände bieten ein wenig Hilfe.

Die Spielfigur wird per Joystick durch die Höhle gesteuert, ein Druck auf den Feuerknopf bringt einen in zahlreiche Menüs, in denen man seine weitere Vorgehensweise bestimmen kann, frei nach dem Motto »Laufen, Zaubern oder Kämpfen«.

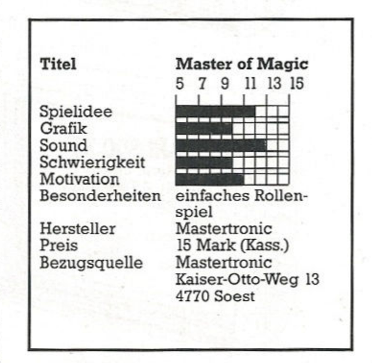
Die Grafik ist auch hier mehr als überdurchschnittlich, der Sound einfach umwerfend. »Master Of Magic« ist wesentlich einfacher zu spielen als »The Last V8« und macht deswegen noch mehr Spaß.

Der schwächste Titel der »M.A.D.«-Spiele ist »Hero Of...«. Hier handelt es sich um ein Action-Adventure, bei dem man fünf Stücke eines Schmuckstückes einsammeln muß. Die Grafik ist recht durchschnittlich, das Spiel hat aber, solange man es nicht gelöst hat, einen gewissen Reiz.

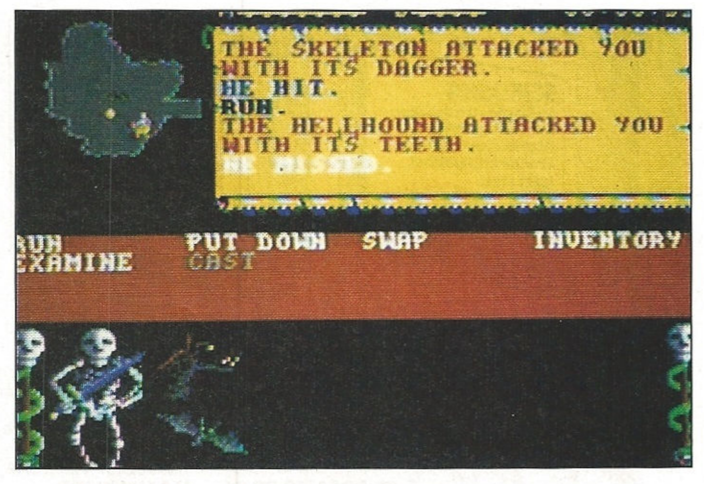
In allerletzter Sekunde erreichte uns die frohe Nachricht, daß in Kürze das Fußballspiel »Five-A-Side-Soccer« als viertes M.A.D.-Spiel herausgegeben wird. »Five-A-Side« erschien schon vor einigen Monaten bei der englischen Firma Anirog. Dies dürfte das einzige Fußballspiel für den C 64 sein, bei dem man sogar foulen und Elfmeter-Schießen kann. Dafür wurden die Einwürfe und Eckstöße nicht berücksichtigt. Eine nette Sprachausgabe unterstützt die Fußball-Simulation, bei der wahlweise zwei oder ein Spieler antreten können.

Bei allen M.A.D.-Spielen gilt auf jeden Fall: Die 15 Mark sind sehr gut angelegt. Was man hier fürs Geld bekommt, sprengt jeden bisherigen Rahmen. Da kann man nur hoffen, daß Mastertronic diesen Qualitätsstandard hält oder sogar ausbaut. Das »Billig-Spiel« ist jetzt jedenfalls salonfähig geworden.

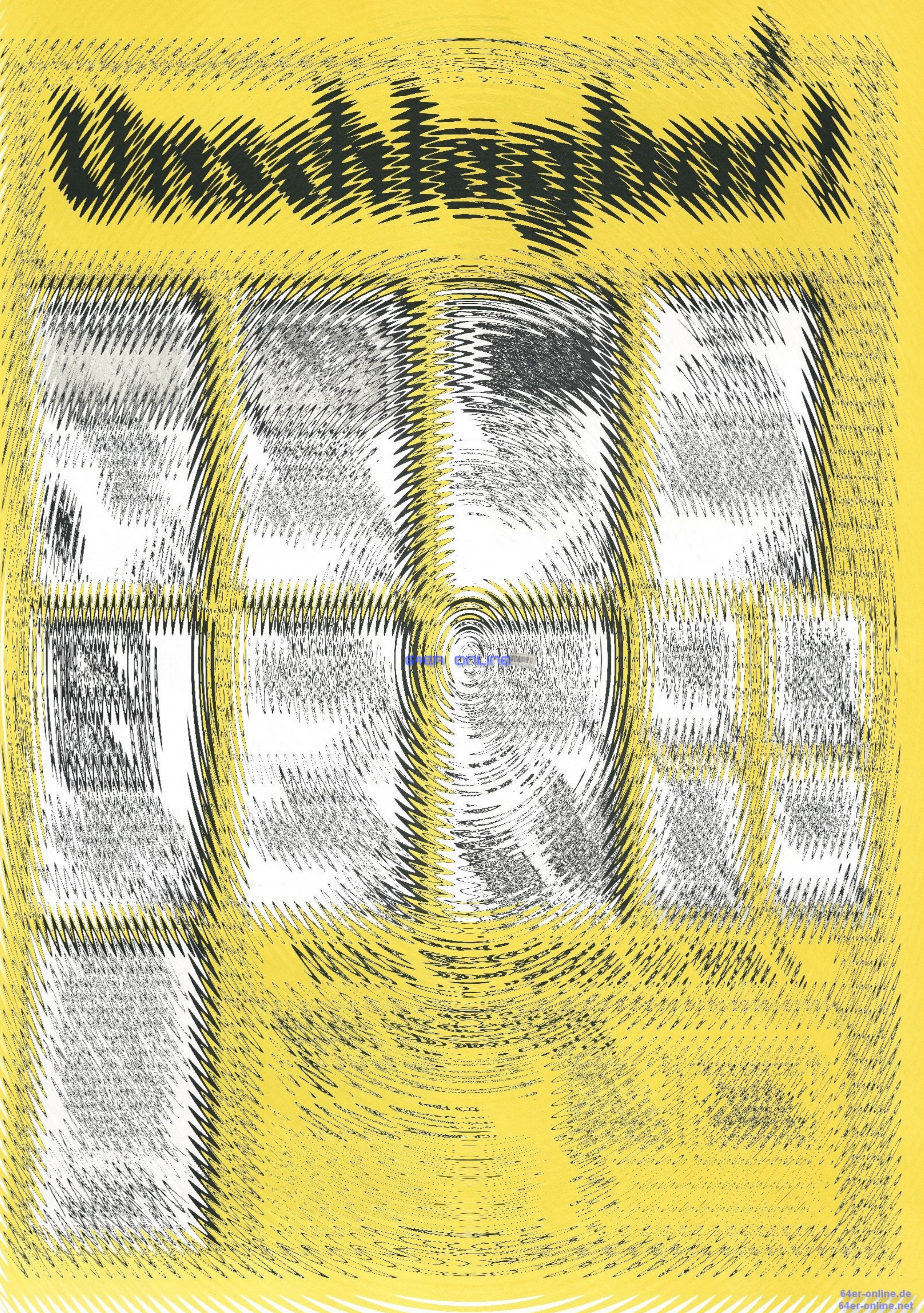
(bs)



Titel	Master of Magic
	5 7 9 11 13 15
Spielidee	
Grafik	
Sound	
Schwierigkeit	
Motivation	
Besonderheiten	einfaches Rollenspiel
Hersteller	Mastertronic
Preis	15 Mark (Kass.)
Bezugsquelle	Mastertronic Kaiser-Otto-Weg 13 4770 Soest



Master Of Magic





Rambo

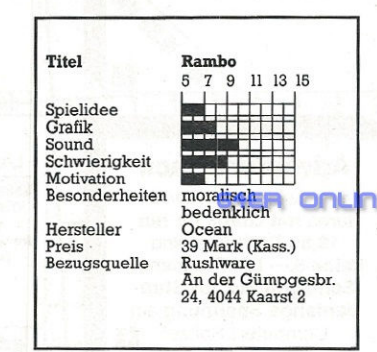
Bildschirmkrieg

Zwei neue Kriegsspiele aus England stürmen die deutschen Bestsellerlisten. »Bei »Rambo« und »Space Invasion« wird geschossen und gemordet, was der Bildschirm hält. Was wir davon halten, lesen Sie hier.

Da kommen aus England zwei neue Spiele, die mit riesigem Aufwand beworben werden. Diese zwei werden auch zu Bestsellern, und gelangen mit Leichtigkeit unter die Top-Ten-Computerspiele im Januar. Alle beide sind mehr oder minder Adaptionen eines der erfolgreichsten Spielautomaten des Jahres 1985. Und beide sind Kriegsspiele...

Es handelt sich um »Rambo« und »Space Invasion«, von denen das letzte eine lizensierte Umsetzung des gleichnamigen Spielautomaten ist. Das Spielprinzip ist recht einfach: Der einsame Super-Soldat muß sich gegen eine Übermacht von bösen Schurken durchsetzen und so viele wie möglich vernichten. Bei »Rambo« wird das Ganze inhaltlich nur dadurch aufgelockert, daß man neben dem Killen der Bösen auch noch die Rettung der Guten übernehmen muß.

»Rambo« hält sich relativ streng an die gleichnamige Film-Vorlage. Diese menschliche Kampfmaschine soll beweisen, daß Vietnamesen immer noch Kriegsgefangenen-Lager unterhalten. Als er selbige findet, mißachtet er seinen Auftrag und unternimmt einen Befreiungsversuch, der natürlich glückt. Daß dabei Hunderte von



Titel	Rambo
	5 7 9 11 13 15
Spielidee	
Grafik	
Sound	
Schwierigkeit	
Motivation	
Besonderheiten	moralisch bedenklich
Hersteller	Ocean
Preis	39 Mark (Kass.)
Bezugsquelle	Rushware An der Gümpgesbr. 24, 4044 Kaarst 2

Gegnern ihr Leben lassen müssen, ist in dem Genre ja selbstverständlich. Im Spiel gibt es fünf verschiedene Phasen: Suche der Gefangenen, Befreiung eines einzelnen, Suche nach einem Helikopter, Befreiung der restlichen Gefangenen und Flucht zur Heimatbasis. Um den Auftrag erfolgreich zu erfüllen, steht Rambo ein ganzes Waffenarsenal zur Verfügung: Maschinengewehr, Pfeil und Bogen, Granate, Hubschrauber, und so weiter. Zwischen diesen Waffen, von denen einige allerdings noch im Gelände aufzusammeln sind, kann er beliebig wechseln. Laute Schüsse rufen nämlich mehr Gegner auf den Plan, so daß sich in der Anfangsphase ein leises Wurfmesser sehr bewährt. Rambo kann sich auf einem sehr großen Spielfeld in alle Richtungen bewegen, das Feld scrollt sehr sauber mit.

Sehr viel gradliniger geht es bei »Space Invasion« zu: Der Spieler soll in ein im Dschungel gelegenes Fort von außerirdischen Gegnern vordringen. Diese sehen dem natürlich nicht tatenlos zu, so daß unser Held, genannt »Super Joe, der Cracksoldat«, reichlich Gebrauch von Maschinengewehr und Handgranate macht. Daß die englischen Programmierer auf den deutschen Markt Rücksicht nehmen wollten, zeigt sich daran, daß das Spiel in England »Commando« heißt, und daß statt Außerirdischer eine irdische Soldaten-Truppe (in verdächtig braunen und grauen Uniformen) das Kampfziel ist. Aber durch Auswechseln von ein paar Sprites macht man aus einem klaren Kriegsspiel kein Unterhaltungs-Programm fürs Kinderzimmer. Denn die »Außerirdischen« sind immer noch recht menschenähnlich und arbeiten nicht mit Laserstrahlen, sondern mit einer »Standard«-Ausrüstung, die vom Granatwerfer bis zum Abfangjäger reicht.

Beide Spiele sind programmiertechnisch kleine Meisterwerke. Das Scrolling und die Animation der Spielfiguren ist sehr gut gelungen. Für die Musik wurden Englands führende Musik-Programmierer engagiert. Martin Galway (Musik sämtlicher Ocean-Programme) schrieb sieben verschiedene Musikstücke für »Rambo« und Rob Hubbard (»Thing on a Spring«) arrangierte eine äußerst packende Version der Spielhallenmusik von »Space Invasion« auf dem C 64.

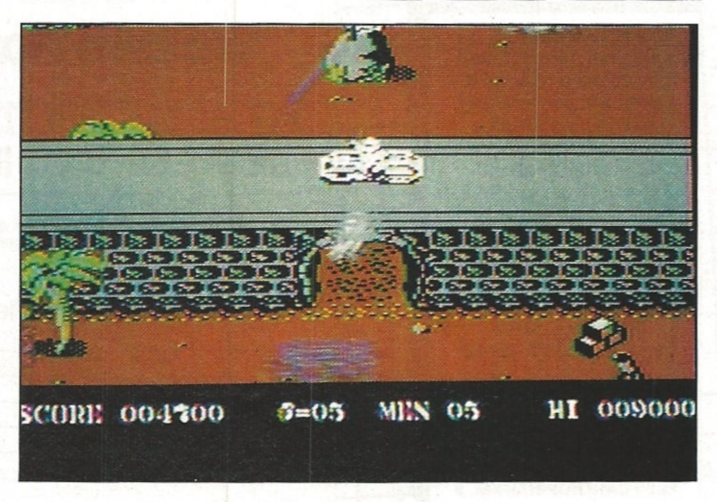
»Space Invasion« wird demnächst auch in einer abgespeckten Version für den C 16 erscheinen.

Ein drittes Spiel dieser Machart geistert ebenfalls durch die Software-Shops: »Who dares wins II« erzeugte in England eine ganze Menge Wirbel — aber nicht aufgrund seiner Handlung. Produzent Alligata verlor einen Rechtsstreit und mußte einige Änderungen am Programm vornehmen, denn es sah »Space Invasion« zu ähnlich. Im Vergleich mit »Rambo« und »Space Invasion« kann »Who dares wins II« nicht mithalten. Grafik und Sound sind mittelmäßig, der Spielwitz noch geringer.

Wenn man diese Spiele sehr genau betrachtet, dann ist man ziemlich böse, daß so viel Programmierer-Energie für ein derartig »mieses« Thema verschenkt wurde. Dachte man schon, daß die Softwarefirmen endlich vom hirnlosen Ballerspiel losgekommen sind, wird man von diesen Neuerscheinungen eines Besseren belehrt. Und die Kunden scheinen diesem Schritt zurück auch noch recht zu geben. (bs)



Titel	Space Invasion
	5 7 9 11 13 15
Spielidee	
Grafik	
Sound	
Schwierigkeit	
Motivation	
Besonderheiten	moralisch bedenklich
Hersteller	Elite Systems
Preis	39 Mark (Kass.)
Bezugsquelle	Rushware An der Gümpgesbr. 24, 4044 Kaarst 2



Space Invasion





Tips und Tricks zu Vizawrite 64 (Teil 4)

Die Resonanz auf unsere Software-Hilfe ist sehr erfreulich. Eine Vielzahl von Tips, Anwendungen und Programmen dazu haben uns erreicht. In dieser Folge geht's in eine neue Runde unserer Software-Hilfe.

Mit Vizawrite 64 kann man ganz hervorragend Listen seiner Disketten herstellen. Mit dem Befehl »CBM und Shift M« schaltet man in den »Merge«-Modus und kann das Directory mit dem »$«-Zeichen einlesen. Nun läßt sich der Directory-Text beliebig bearbeiten und mit Kommentaren versehen.

Auf diese Weise kann man mehrere Directorys aneinanderhängen und als Textdatei speichern. Wenn sich auf einer Diskette etwas ändert, wird einfach das gesamte geänderte Directory mit Vizawrite 64 in das bestehende Verzeichnis geladen. Dann übernimmt man die entsprechenden Kommentare und löscht den alten Inhalt mit der Taste F7. Auf diese Weise steht immer der aktuelle Inhalt aller Disketten zur Verfügung. Zum Ausdrucken empfiehlt es sich, im Druckermenü den »Pitch Setting«-Wert auf »3« (komprimierte Schrift) und den Zeilenabstand auf 1/8 Zoll einzustellen. Als Startspalte wird »0, 45 oder 80« eingegeben. Das bewirkt, daß man ein A4-Blatt 3spaltig bedrucken kann. Der Ausdruck der Programme wird dadurch sehr übersichtlich und man muß nicht durch viele Seiten blättern um ein Programm zu finden (Bild 1).
Dieser Tip stammt von Horst Kneisel.

Umlaute auf dem MPS 801

Hendrik Hartje hat eine Anpassung entwickelt, mit der die Druckroutine für den MPS 801 aus der Ausgabe 2/86, Seite 75, auch mit Vizawrite 64 Umlaute auf dem MPS 801 druckt. Dazu geht man wie folgt vor:
1.) Die Druckroutine mit »LOAD"Druckroutine",8« laden und starten.
2.) Bei der Bereichswahl »5« für Basic-Ende angeben.
3.) Auf die Frage » Zeichen ändern?« ein »J« für Ja eingeben.
4.) Für die Umlaute müssen nun die folgenden Daten angegeben werden:

ä = 123
Ä = 91
ö = 124
Ö = 92
ü = 125
Ü = 93
ß = 126

5.) Das vom Programm vorgegebene NEW und PRINT FRE(0) mit RETURN übernehmen.
6.) OPEN 4,4,7:PRINT#4:CLOSE 4 eingeben um den Drucker auf die neue Schrift einzustellen.
7.) Das Programm Vizawrite 64 normal laden und den gewünschten Text mit Umlauten tippen. Zur Erleichterung kann man sich kleine Aufkleber auf die Tasten kleben, um die Umgewöhnung auf die Umlaute zu erleichtern: Auf dem Semikolon das »Ä«, auf dem Doppelpunkt das »Ö«, auf den Klammeraffen das »Ü« und auf das Pfund-Zeichen das »ß«.
8.) Zum Drucken im Druckermenü als Druckertyp »y« angeben und den Druckvorgang mit Taste F1 starten.

Achtung! Durch die neue Schrift passen weniger Zeilen auf ein Blatt (etwa 55). Dies muß unbedingt beachtet werden, da der Ausdruck sonst über die Perforation reichen würde.

Jetzt können Sie Ihren ersten MPS-gedruckten Text mit Umlauten und Unterlängen bewundern.

(Hendrik Hartje)

Fortsetzung auf Seite 162

```
"64'er 10/84      " bo0
"apocalypse now"   prg   71
"licht-tele(4baud" prg   22
"hypra load"       prg    6
"disk-dump"        prg   19
"flist"            prg    3
"maske datas"      prg   32
"supercopy"        prg   12
"demo/maske"       prg    5
"menue"            prg    5
"usertastatur"     prg   16
"funkt.tast"       prg    9
"Zinsrechnung"     prg   19
"Zinseszinsrechnu" prg   20
"Diskontieren ein" prg   19
"Erstellen eines " prg   27
"hc-gemini10xx"    prg    3
"hc-mps 801"       prg    7
"hc-vc 1526"       prg    7
"olymp.-comp.-2"   prg   26
"hc-epson"         prg    5
"hc 1520 farbig"   prg   18
"-------vc-20----" usr    1
"diskorganisat.64" prg   42
"progr.listen 1"   prg    1
"progr.listen 2"   prg    1
"progr.listen 3"   prg    2
"progr.listen 4"   prg    1
"video-vorspann"   prg   12
"----------------" usr    1
```

```
"einz. scroll not" prg    1
"einz.save masch." prg    2
"einz.vc20-klavie" prg    1
"einz.microsound"  prg    1
"einz. error 21"   prg    1
"einz. merge"      prg    1
"einz.rechtsbuend" prg    1
"-befehle-simons-" seq    1
"list 1"           prg    2
"list 2"           prg    1
"list 1+2"         prg    3
"list 3"           prg    2
"list 4 (error)"   prg    3
"list 5 (joy)"     prg    3
"list 6 (dump)"    prg    3
"list 7 (page)"    prg    2
"list 4+5+6+7"     prg    9
"-----vc-20------" usr    1
"pseudosprts.data" prg   20
"spritegenerator"  prg   19
"sprite-demo"      prg    1
"laterna magica"   prg    5
"laterna ed"       prg    7
"erweiterung"      prg   17
"supergrafik"      prg    4
"---vc-20-kurs---" seq    1
"data erzeuger"    prg    5
"funktionstasten"  prg   10
"befehlskopf"      prg   16
"token erzeuger"   prg    3
blocks free.            159
```

```
"alu1.pic"         prg   33
"alu2.pic"         prg   33
"alu3.pic"         prg   33
"alu4.pic"         prg   33
"alu5.pic"         prg   33
"alu6.pic"         prg   33
"demo2.pic"        prg   33
"art.pic"          prg   33
"baby.pic"         prg   37
"----------------" prg    1
"form.eingabe"     prg   19
"----------------" del    0
"hypra.load-track" prg   37
"hypra-kernal"     prg   26
"led-test/12"      prg    2
"und-generator"    prg    1
"auto-&-maker"     prg    3
"----------------" del    0
"c/quicktext"      prg   53
blocks free.             33

"64/01/84        " bw0    0
sicherheitskopie ..........

"64'er 5/85      " 05-
"checksum.schnell" prg    5
"mse lader"        prg   29
"mse v1.0"         prg    7
"----------------" del    0
"mps802.listing2"  prg    2
```

So sieht eine Directory-Liste nach dem Ausdruck aus.

Die dreifache Gewinnchance
exklusiv für alle »64'er«-Abonnenten:

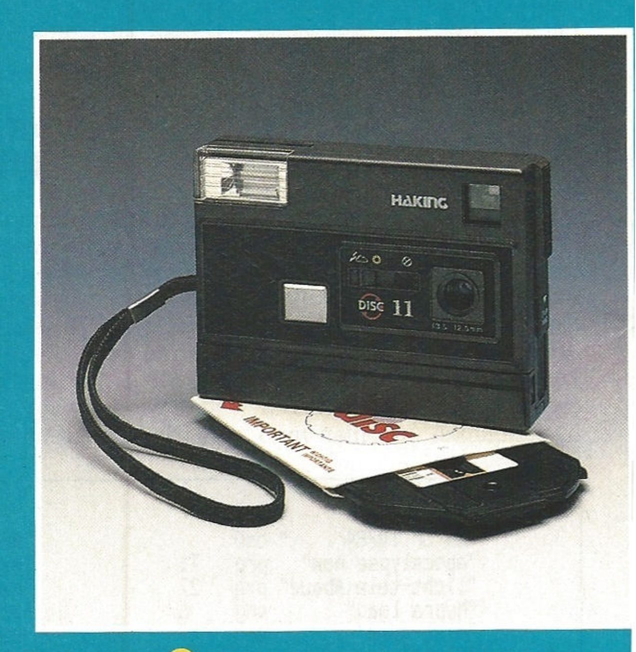
VON IHNEN NEUEN 64'er-ABONNENTE EINE TOLLE P



Prämie Nr. 1 für einen geworbenen Abonnenten

Calculator-Watch

5 Zeitfunktionen der LCD-Uhr und dazu ein Rechner mit allen wichtigen Rechenfunktionen auf kleinstem Raum. Am Handgelenk, Schlüsselbund oder am Gürtel zu tragen. LCD-Uhr mit Anzeige für Sek., Min., Std., Tag, Monat und zusätzlicher Alarmfunktion.



Prämie Nr. 2 für zwei geworbene Abonnenten

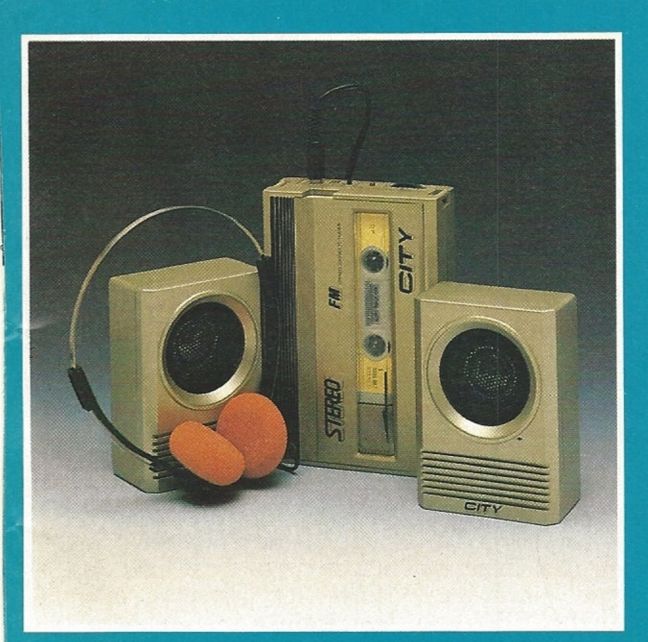
Disc-Kamera mit eingebautem Blitz und Film

Die »Schnappschuß«-Kamera in flacher handlicher Form, allzeit griffbereit durch die praktische Trageschlaufe. Disc-Kamera mit eingebautem Blitz für die Verwendung von Color-Disc-Filmen, 15 Bilder. Blende f. 3,5. Objektiv 12,5 mm.

*** Mit jedem geworbenen, neuen Abonnenten steigern Sie den Wert Ihrer Prämie!**

Benutzen Sie zur Anforderung Ihrer Prämie den nebenstehenden Bestellabschnitt.

FÜR JEDEN GEWORBENEN N BEKOMMEN SIE R Ä M I E*



Prämie Nr. 3 für drei geworbene Abonnenten

Stereo-Cassetten-Recorder mit Radio und Boxen

Solo-Musik-Jump mit Stereo-Kopfhörer oder geselliger Party-Sound mit zwei Mini-Stereo-Boxen. Das Cassetten-Abspielgerät *mit eingebautem Radioteil* ermöglicht, Top-Hits auf Cassette zu hören oder den aktuellsten News auf UKW zu folgen. Radio mit LED-Stereo-Anzeige. UKW-Stereo-Mono-Schalter, Kopfhörerbuchse, Kopfhörer und Trageriemen. Recorder-Funktionen: Play, schneller Vorlauf, Stop, Eject. FTZ-geprüft.

Erster NEUER 64'er-Abonnent

Ich abonniere »64'er« zum nächstmöglichen Termin. Ich beziehe »64'er« bisher noch nicht regelmäßig und möchte die Vorteile eines persönlichen Abonnements nutzen.

Ich bezahle einschließlich Frei-Haus-Lieferung für 12 Hefte DM 78,– jährlich im voraus.

Die Bestellung gilt mindestens ein Jahr und weiter bis zur Abbestellung. Das Abonnement verlängert sich automatisch um ein weiteres Jahr zu den dann gültigen Bedingungen, wenn es nicht 2 Monate vor Ablauf schriftlich gekündigt wird.

Liefer- und Rechnungsanschrift:

Name/Vorname

Straße

PLZ/Ort

Datum/Unterschrift

Mir ist bekannt, daß ich die Bestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen kann. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.
Ich bestätige dies durch meine 2. Unterschrift.

Datum/Unterschrift

Zweiter NEUER 64'er-Abonnent

Ich abonniere »64'er« zum nächstmöglichen Termin. Ich beziehe »64'er« bisher noch nicht regelmäßig und möchte die Vorteile eines persönlichen Abonnements nutzen.

Ich bezahle einschließlich Frei-Haus-Lieferung für 12 Hefte DM 78,– jährlich im voraus.

Die Bestellung gilt mindestens ein Jahr und weiter bis zur Abbestellung. Das Abonnement verlängert sich automatisch um ein weiteres Jahr zu den dann gültigen Bedingungen, wenn es nicht 2 Monate vor Ablauf schriftlich gekündigt wird.

Liefer- und Rechnungsanschrift:

Name/Vorname

Straße

PLZ/Ort

Datum/Unterschrift

Mir ist bekannt, daß ich die Bestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen kann. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.
Ich bestätige dies durch meine 2. Unterschrift.

Datum/Unterschrift

Dritter NEUER 64'er-Abonnent

Ich abonniere »64'er« zum nächstmöglichen Termin. Ich beziehe »64'er« bisher noch nicht regelmäßig und möchte die Vorteile eines persönlichen Abonnements nutzen.

Ich bezahle einschließlich Frei-Haus-Lieferung für 12 Hefte DM 78,– jährlich im voraus.

Die Bestellung gilt mindestens ein Jahr und weiter bis zur Abbestellung. Das Abonnement verlängert sich automatisch um ein weiteres Jahr zu den dann gültigen Bedingungen, wenn es nicht 2 Monate vor Ablauf schriftlich gekündigt wird.

Liefer- und Rechnungsanschrift:

Name/Vorname

Straße

PLZ/Ort

Datum/Unterschrift

Mir ist bekannt, daß ich die Bestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen kann. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.
Ich bestätige dies durch meine 2. Unterschrift.

Datum/Unterschrift

Vermittler:

Ich bin bereits Abonnent des »64'er Magazin« und habe nebenstehende(n) Abonnenten für Sie geworben.

Ich erhalte von Ihnen

für einen geworbenen Abonnenten die Prämie
☐ Nr. 1
für zwei geworbene Abonnenten die Prämie
☐ Nr. 2
für drei geworbene Abonnenten die Prämie
☐ Nr. 3

sofort nach Eingang der Abonnenten-Zahlung(en).
Ich weiß, daß Eigenwerbung ausgeschlossen ist.
Bitte schicken Sie die Prämie an meine Adresse:

Name/Vorname

Straße

PLZ/Ort

Datum/Unterschrift

Markt&Technik
ZEITSCHRIFTEN
VERLAG

Bestellabschnitt ausfüllen, ausschneiden und im Kuvert einsenden an: »64'er« Leser-Service, Postfach 1304, 8013 Haar

64486

Fortsetzung von Seite 159

Ganz besonders fleißig war Rolf Kitzing, denn er hat uns gleich vier gute Tips eingeschickt:

Bestimmung der ersten Seitenzahl

Vizawrite 64 verfügt über keinen Befehl, mit dem man für den Druck eines Textes willkürlich die Seitenzahl bestimmen kann, mit der die Seitennumerierung beginnen soll. Eine solche Möglichkeit wäre zum Beispiel für den Ausdruck längerer Texte mit automatischer Seitennumerierung, die sich über mehr als eine Diskette erstrecken, von Nutzen. Wenn die Textkette der ersten Diskette sich zum Beispiel von Seite 1 bis Seite 20 erstreckt, dann müßte die Numerierung der zweiten Textkette bei Seite 21 beginnen. Vizawrite 64 würde aber erneut bei »1« zu zählen beginnen. Normalerweise müßten also alle Seitenzahlen von Hand eingefügt werden, wenn dieses Problem vermieden werden soll. Es gibt jedoch eine, wenn auch etwas umständliche Möglichkeit, dennoch eine Numerierung zu veranlassen, die nicht automatisch bei »1«, sondern zum Beispiel bei »21« beginnt. Im Beispielfall brauchen vor der ersten Seite des Textes nur zwanzig Seitenendezeichen (CTRL p) eingegeben werden. Dadurch wird die eigentliche Seite 1 des Textes zur Seite 21 von X Seiten. Wenn dann auf der Druckerauswahlseite unter »Start Page« »21« und unter »End Page« »999« eingegeben wird, beginnt Vizawrite 64 die Numerierung der Seiten der zweiten Textkette mit »21«. Auf diese Weise kann jede gewünschte Anfangsseitenzahl gewählt werden.

Unter »Global/Fill« darf bei diesem Verfahren allerdings kein »f« stehen, weil Vizawrite 64 dann keinerlei Aktion veranlaßt.

Für kontinuierliche Mehrfachdrucke stellt Vizawrite 64 leider keine Funktion bereit. Trotzdem kann man mit einem Druckvorgang zwei oder mehr Exemplare eines Textes auf dem Umweg über die »Merge«-Funktion erhalten.

Wenn es sich um einen einseitigen Text handelt, gibt man dazu in die Arbeitsseite so viele Einfügezeichen (CTRL m) ein, wie Ausdrucke gewünscht werden. Hinter jedes dieser Einfügezeichen, ausgenommen dem letzten, wird ein Nichteinfügezeichen (CTRL d) gesetzt. Falls zum Beispiel drei gedruckte Exemplare des bearbeiteten Textes gewünscht werden, wäre folgendes einzugeben:

Mehrfachdrucke

CTRL m/CTRL d/CTRL m/
CTRL d/CTRL m/RETURN

In den Haupttext wird an den Anfang der ersten Zeile ebenfalls ein Einfügezeichen gesetzt.

In die Druckauswahlseite wird dann unter »Global/Fill« ein »f« eingegeben und wenn nun der Druckbefehl erteilt wird, produziert der Drucker genau so viele Exemplare des Textes nacheinander ohne Unterbrechung wie Einfügezeichen in der Arbeitsseite stehen und ohne daß der Druckbefehl erneut erteilt werden muß.

Bei diesem Verfahren ist nur darauf zu achten, daß die automatische Seitennumerierung nicht benutzt wird, da Vizawrite 64 dann zum Beispiel die Mehrfachdrucke eines einseitigen Textes nicht, wie es richtig wäre, alle mit »1« numeriert, sondern das zweite Exemplar mit »2«, das dritte mit »3« und so weiter. Der mehrfache fortlaufende Druck eines aus nur einer Seite bestehenden Textes ist, wie man sieht, verhältnismäßig einfach. Etwas komplizierter wird es, wenn man auch mehrseitige Texte in einem Zug in mehreren Exemplaren ausdrucken will. Möglich ist aber auch das und zwar kann man dabei je nach Geschmack auf dreierlei Art und Weise vorgehen.

Falls man auch hier davon ausgeht, daß drei gedruckte Exemplare des mehrseitigen Textes gewünscht werden, dann müßte das oben beschriebene Verfahren für einseitige Texte ja eigentlich zum gleichen Resultat führen. Aber leider hat Vizawrite 64 die Eigenheit, beim mehrfachen Druck mehrseitiger Texte auf diese Weise vom jeweils letzen Ausdruck, hier also vom dritten, nur die erste Seite zu produzieren und alle folgenden Seiten unter den Tisch fallen zu lassen. Aber auch gegen diesen Tick des Programms, der sich übrigens auch bei der Erstellung von Serienbriefen bemerkbar macht, ist ein Kraut gewachsen. Entweder gibt man in die Arbeitsseite ein Einfügezeichen mehr ein als man Ausdrucke benötigt, dann bekommt man drei vollständige Texte plus noch einmal die unerwünschte erste Seite des Textes zum Wegwerfen. Wer diese Papierverschwendung unerfreulich findet, der kann sie durch Anwendung der dritten, etwas umständlicheren Methode aus der Welt schaffen. Dieses dritte Verfahren veranlaßt den Drucker, ganz genau nur die gewünschte Anzahl vollständig gedruckter Texte auszugeben.

Um dies zu erreichen, werden in die Arbeitsseite nacheinander so viele Einfügezeichen eingegeben, wie der zu druckende Text Seiten hat. Nehmen wir an, unser Text hat drei Seiten und soll zweimal gedruckt werden. Das erfordert also erstmal die Eingabe von drei Einfügezeichen nacheinander. Gleich danach wird ein Nichteinfügezeichen gefolgt von erneuten drei Einfügezeichen sowie einem abschließenden »RETURN« angefügt.

Die Befehlsfolge lautet also genau wie folgt:

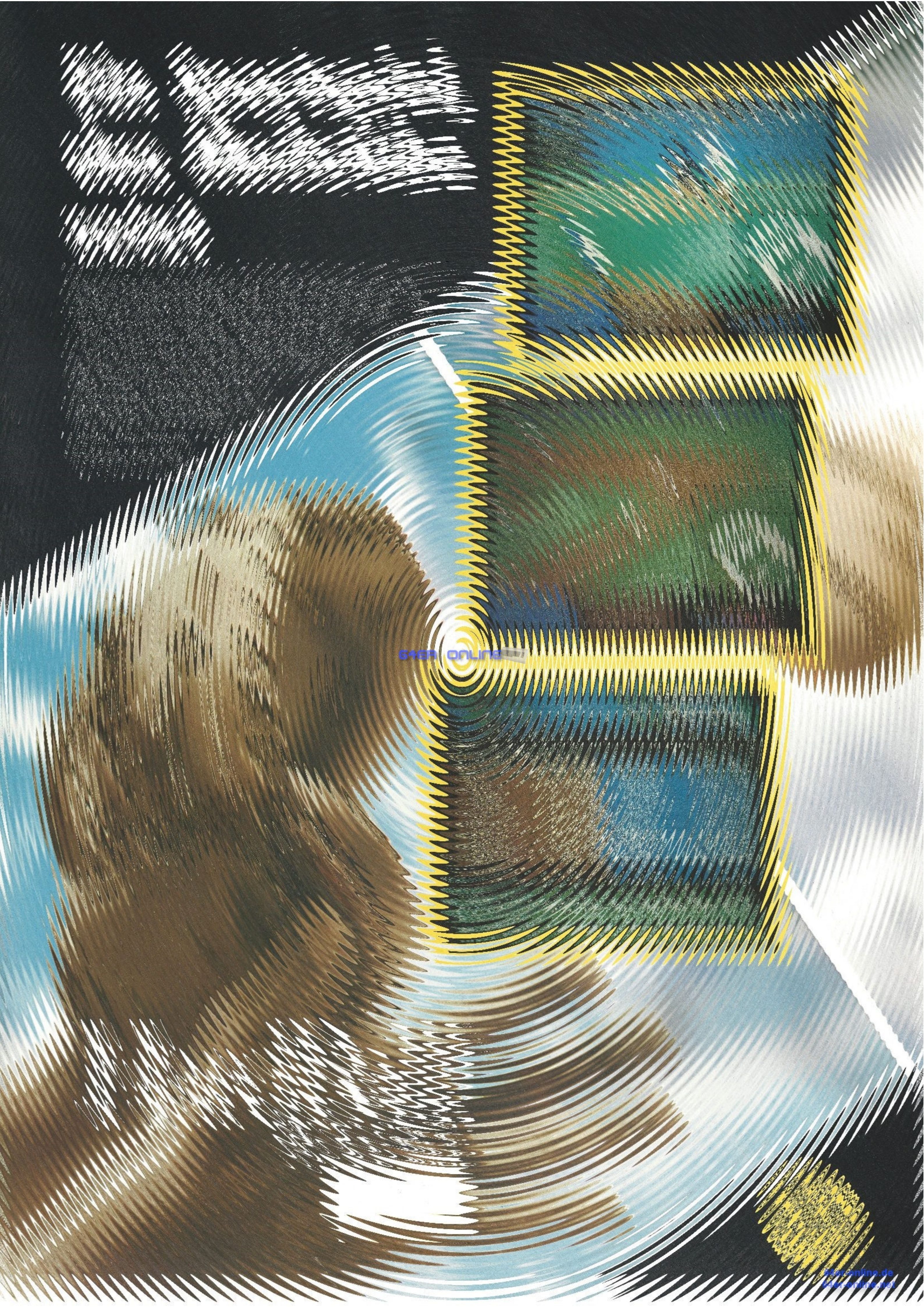
CTRL m/CTRL m/CTRL
m/CTRL d/CTRL m/ CTRL
m/CTRL m/»RETURN«.

Jedes der einzelnen Einfügezeichen in einer der durch Nichteinfügezeichen voneinander getrennten Zeichenreihen steht hierbei für eine Seite des mehrseitigen Textes. Besteht der Text, von dem mehrere Ausdrucke gewünscht werden, aus vier oder fünf Seiten, so müßten entsprechend der Seitenzahl vier oder fünf Einfügezeichen in eine Reihe gesetzt werden. Jede der durch Nichteinfügezeichen voneinander getrennten Reihen von Einfügezeichen steht für einen Druck. Werden also nicht zwei, sondern drei oder vier Drucke gewünscht, so müßte die Befehlsreihe nur entsprechend verlängert werden. Um nach diesen Vorbereitungen endlich die erwünschte Wirkung zu erreichen, muß nur noch an den Anfang der ersten Zeile einer jeden Seite des betreffenden Textes ein Einfügezeichen und nach »Global/Fill« in der Druckerauswahl ein »f« gesetzt werden. Setzt man die Einfügezeichen beim Schreiben des Textes gleich mit ein, weil möglicherweise mehrere Ausdrucke benötigt werden, ist der Arbeitsaufwand dafür gering. Sollte am Ende dann doch nur ein Ausdruck erforderlich sein, kann man die Einfügezeichen unbesorgt für späteren Bedarf stehen lassen, weil man den ursprünglich geplanten Mehrfachdruck durch einfaches Auslassen des »f« in der Druckerauswahl außer Kraft setzt.

Ein letzter Hinweis zu diesem Thema: Bei der Verwendung von Hochschrift oder Tiefschrift in einem auf diese Weise mehrfach gedruckten Text ist es besser, diese Schriftarten nicht mit den Vizawrite 64-Formatzeichen, sondern besser mit dem Drucker-Steuerbefehl über die Formatzeile einzustellen, weil Vizawrite 64 nach dem Ausdruck des ersten Exemplars unter gewissen Umständen die Einstellung der Hoch- oder Tiefschrift nicht beachtet.

Fettdruck mit Epson

Bei der Verwendung eines Epson-Druckers mit NLQ-Schrift stellt sich heraus, daß der bei Vizawrite 64 verfügbare Fettdruckbefehl (CTRL e) eigentlich ein Befehl ist, der Doppeldruck und nicht Fettdruck ein- und ausschaltet. Man merkt dies bei der genauen Beobachtung des Normalschrift-Druckvorgangs oder spätestens, wenn man sich wundert, warum der Befehl beim Druck von NLQ-Schrift keine Wirkung zeigt. Letzteres liegt natürlich daran, daß die NLQ-Schrift keinen Doppeldruck, wohl aber Fettdruck zuläßt. Um nun auch bei der NLQ-Schrift den gewünschen Effekt zu erzielen, bedient man



sich am besten der Steuerbefehle, indem man sich mit »E« und »F« in der Formatzeile die Einschaltung und Ausschaltung des Fettdrucks programmiert.

Worttrennung am Ende der Zeile

Bei Druck im Blocksatz vermeidet man gern allzu lange Lücken in einer Zeile, die dadurch entstehen, daß das erste Wort der folgenden Zeile zu lang ist, um noch ganz ans Ende der Vorzeile zu passen. Es wäre also eine entsprechende Trennung dieses Wortes angezeigt, so daß wenigstens ein Teil des Wortes noch in die Vorzeile passen und damit die bestehende Lücke entsprechend verkleinert würde.

Nun verfügt Vizawrite 64 aber leider nicht über die angenehme Funktion des bedingten Trennungsstriches. Man muß sich also anders helfen und kann das auch eigentlich ziemlich gut, indem man wie folgt verfährt:

Nachdem der eingegebene Text in jeder anderen Hinsicht für richtig befunden worden ist, wird der Cursor auf den rechten Zeilenrand gestellt und der rechte Rand des Textes von oben nach unten auf bestehende Unterschiede in der Zeilenlänge überprüft. Findet man eine Zeile, die besonders kurz geraten ist, kann man daraus schließen, daß sich hier beim Ausdruck im Blocksatz besonders auffallende Zwischenräume ergeben werden. Wenn nun das erste Wort der folgenden Zeile trennbar ist, kann man diese Zwischenräume durch Herüberziehen des Wortanfangs an das Ende der Vorzeile unter Umständen bedeutend verringern oder gar ganz verschwinden lassen.

Um diese Wirkung zu erzielen, verfährt man wie folgt: Der Cursor muß hinter der kurzen Zeile stehen. Dann wird SHIFT und RETURN gedrückt und der Cursor damit an den Anfang der folgenden Zeile bewegt. Von dort setzt man ihn dann mit der Cursor-Steuertaste auf den Buchstaben des ersten Wortes, der auf die Silbe oder die Silben folgt, die noch in der vorhergehenden Seite Platz finden, das heißt, also auf den ersten Buchstaben des Restwortes, das auf der folgenden Zeile verbleiben soll. Durch Drücken von SHIFT und INST/DEL wird der Wortteil, der vor dem Cursor steht, nun vom Rest des Wortes getrennt und automatisch ans Ende der vorhergehenden Zeile übertragen. Der Cursor springt mit und es muß jetzt nur noch ein Trennungsstrich an das Ende des versetzten Wortteils gefügt werden. Das Restwort in der folgenden Zeile rückt automatisch an den linken Rand. Da der hier verwendete Trennungsstrich aber leider wie gesagt kein bedingter Trennungstrich ist, muß man nach dem Abschluß dieser Textjustierung vorsichtig mit nachträglichen Änderungen sein. Da Vizawrite 64 bekanntlich den Text dann sofort umformatiert, kann es natürlich sein, daß dadurch ein getrenntes Wort vom Zeilenende in die Mitte einer Zeile rutscht und dann beim Ausdruck mit einem etwas unmotivierten Trennungsstrich erscheint. Nach vorgenommenen nachträglichen Textänderungen sollte man also vorsichtshalber immer den unformatierten Text vor dem Druck auf solche Erscheinungen überprüfen.

Tips fürs Drucken

In der Formatzeile können viele Steuerzeichen für den Drucker abgekürzt werden, indem ihr ESC-Buchstabe geschrieben wird. Der Grund ist einfach, denn es werden von Vizawrite 64 nur die Zeichen die kleiner als 128 sind unbehandelt weitergegeben, die ASCII-Zeichen über 128 werden folgendermaßen verarbeitet: ASCII-Wert — 128 = zu druckendes Zeichen oder einfach »AND $8F«, davor wird der Code 27 (der Wert, der viele Steuerzeichen einleitet, heißt ESC und hat den Wert 27) gesetzt. Dies bewirkt, daß zum Beispiel der Code zum Ausschalten des Potenzmodus (27,84) durch ein »CTRL 0 = T« in der Formatzeile abgekürzt werden kann. Wichtig ist, daß ein großes »T« dort steht — im Commodore Bildschirmcode ist das der Wert 128 + 84 = 212.

In vielen Fällen ist das ganz praktisch, es hat aber Nachteile bei der Verwendung von etwas »exotischeren« Druckern, die Steuercodes mit Werten über 128 verwenden, so zum Beispiel verschiedene Typenraddrukker. Abhilfe: Eine Schnittstelle schreiben, die diese umgewandelten Zeichen erkennt und zurückverwandelt! (Andreas Gebbert)

Neues von Vizaspell

In der letzten Folge stellten wir die Frage nach dem Rechtschreibprogramm Vizaspell, das ja von Vizawrite 64 aus aufgerufen werden kann. Dazu konnten wir in Erfahrung bringen, daß man dieses Programm zum Preis von 129 Mark bei Microtron in der Schweiz bekommen kann. Leider ist dies nur die englische Version, die bei Verwendung von deutschen Umlauten abstürzt und auch nur über einen englischen Wortschatz verfügt. Es besteht allerdings die Möglichkeit, beliebige Worte zusätzlich einzuprogrammieren, also auch deutsche Worte (ohne Umlaute). Trotzdem, wer mehr über Vizaspell weiß, sollte sich bitte melden. (aw)

Vizawrite 64 und Privileg Electronic 3000

Die Privileg Electronic 3000 mit »Multi Board Interface« hat leider die unangenehme Eigenschaft, kleine und große Buchstaben miteinander zu vertauschen. Das kann man aber abstellen. Durch einmaliges Senden des CHR$(145) schaltet der Drucker um und schreibt von nun an ganz korrekt. Diesen Befehl muß man auf jeder Seite neu eingeben.

Vorgehensweise: In der Formatzeile die CTRL-Taste drücken, dann eine Zahl von 0 bis 9, danach » = 145« eingeben. In der ersten Textzeile den neuen Formatbefehl als erstes angeben. Beim Ausdruck hat sich das »a« für den Druckertyp am besten bewährt.

Einen beliebigen Text kann man mit dem CHR$(95) in der Formatzeile so oft wie erforderlich erreichen.

(Karl-Heinz Nöthel)

Aussichten

In der nächsten Folge gibt es Programme, mit denen das Arbeiten mit Vizawrite 64 noch interessanter wird. Das erste Programm liest beliebige Vizawrite 64-Dateien ein und gibt sie entweder auf dem Bildschirm oder in eine sequentielle Datei aus. Mit dem zweiten Programm erhalten Sie einen Zeichensatz-Editor, der es ermöglicht, beliebige Zeichen zu definieren und mit Vizawrite zu verwenden. Seien es nun griechische oder wissenschaftliche Zeichen — der Fantasie sind keine Grenzen gesetzt.

Die übernächste Folge wird neben vielen weiteren Tips noch ein ganz besonderes Programm enthalten. Mit diesem Programm können Sie mit Vizawrite 64 Assemblerfiles erstellen und anschließend assemblieren. Dabei stehen Ihnen alle Vorteile eines so leistungsfähigen Texteditors wie Vizawrite 64 zur Verfügung. (aw)

In eigener Sache.

Immer wieder erreichen mich Briefe mit der Bitte, aus dem Software-Corner doch eine Serie zu machen und auch andere Programme wie Superbase, Startexter, Vizastar 64, Datamat, Multiplan und Textomat zu behandeln. Nun, diesen Wunsch will ich gerne erfüllen. Dabei möchte ich, nachdem in einer oder zwei Folgen auf die generelle Bedienung des Programmes eingegangen wurde, Ihre Tips, Tricks, Erweiterungen und Anwendungen aber auch Fragen in lockerer Folge veröffentlichen. Wer sich beteiligen möchte, und auch den anderen Lesern seine Erfahrungen zugute kommen lassen will, schreibt unter dem Stichwort »Software Corner« an den Verlag Markt & Technik AG, Arnd Wängler, Redaktion 64'er, Hans-Pinsel-Str. 2, 8013 Haar.



Superbase

(Teil 1)

Nach den Tips zu Vizawrite sollen nun Anwenderprobleme mit Superbase 64 besprochen werden. Hier ist die erste Folge.

Bei Superbase handelt es sich um ein Datenbankprogramm mit äußerst vielfältigen Möglichkeiten. Superbase ist derart komplex, daß Wochen intensiver Arbeit notwendig sind, um alle Möglichkeiten zu erkennen. Doch sollte sich niemand dadurch abschrecken lassen, denn durch die perfekt zugeschnittene Menütechnik kann Superbase nach kurzer Vorbereitungszeit als Dateiverwaltungssystem verwendet werden.

Die Vielfältigkeit von Superbase liegt darin, daß der Anwender die Möglichkeit hat, immer wiederkehrende Befehlssequenzen zu einem Programm zusammenzufassen und dadurch letztlich seine Anwendung zu vereinfachen. Er kann sogar eigene Menüs erstellen.

Help — eine gute Hilfe

Zu Beginn der Arbeit mit Superbase empfiehlt es sich, entsprechend der Systemmeldung eine Datendiskette zu erstellen. Durch diesen Vorgang wird die Diskette formatiert und die HELP-Bildschirme sowie das Startprogramm werden auf die Datendiskette kopiert. Im Inhaltsverzeichnis der Diskette — aus dem 2. Menü über MAINTAIN zu erreichen — lassen sich die HELP-Bildschirme an einem vorangestellten »h« erkennen. Alle Programme sind am Ende des Namens an ».p« erkennbar. Dieses »h« darf im HELP-Modus nicht mit eingegeben werden.

Den Inhalt der aufgerufenen Bildschirme kann man durch gleichzeitiges Drücken von »CTRL« und »P« ausdrucken lassen. Unter Umständen bestehen sie aus mehreren Seiten, die nacheinander aufgerufen werden müssen. Jedem Anfänger mit Superbase sei empfohlen, sich diese Hilfen in einer ruhigen Minute durchzulesen.

Datenbank wird eröffnet

Nach dem erneuten Starten von Superbase werden Sie aufgefordert, die Datendiskette einzulegen und dies mit RETURN zu bestätigen. Anschließend läuft das START-Programm an und fragt Sie nach dem Namen der Datenbank. Geben Sie zum Beispiel »personen« ein. Da eine Datenbank solchen Namens noch nicht existiert, wird sie nach erfolgter Bestätigung angelegt. Nun fragt das System nach einer Datei. Hier müssen Sie wieder einen Namen eingeben; zum Beispiel »adressen«. Da diese Datei auch noch nicht besteht, muß sie durch Bestätigung der Abfrage angelegt werden. Auf diese Weise gelangt man automatisch in den FORMAT-Modus.

Nun bekommt die Sache Format

Hier muß das Datensatz-Layout angelegt werden. Dafür stehen dem Anwender bis zu vier Bildschirmseiten zur Verfügung. Doch bevor das Layout erstellt wird, müssen einige wenige Punkte erläutert werden. Im FORMAT-Modus wird, wie bei jedem anderen Dateiverwaltungsprogramm, die Länge der einzelnen Datensatzfelder festgelegt. In der Länge der Felder sind dem Anwender von Superbase kaum Grenzen gesetzt. Er darf die Datensatzlänge von 1108 Zeichen und die Anzahl von 127 Feldern nicht überschreiten.

Damit Sie später auch wissen, was in die einzelnen Felder eingetragen werden soll, bekommen diese einen Namen. Hierbei müssen Sie eines beachten: Superbase rechnet alle Zeichen bis zum nächsten Blank, die vor dem Feld stehen, zum Feldnamen. »1. Stichwort« und »2. Stichwort« dürfen Sie folglich nicht als Feldnamen angeben. Sie haben jetzt zwei Felder mit dem Namen »Stichwort«, obwohl in dem Layout ja etwas anderes steht. Die Zahlen werden als Notizen festgehalten, gehören jedoch nicht zum Feldnamen.

Der Schlüssel zur Datenbank

Jeder Datensatz muß bei Superbase ein Schlüsselfeld erhalten. Es dient dem Programm dazu, die Datensätze in alphabetischer Reihenfolge abzuspeichern. Außer-

Tabelle 1. Steckbrief: Superbase 64

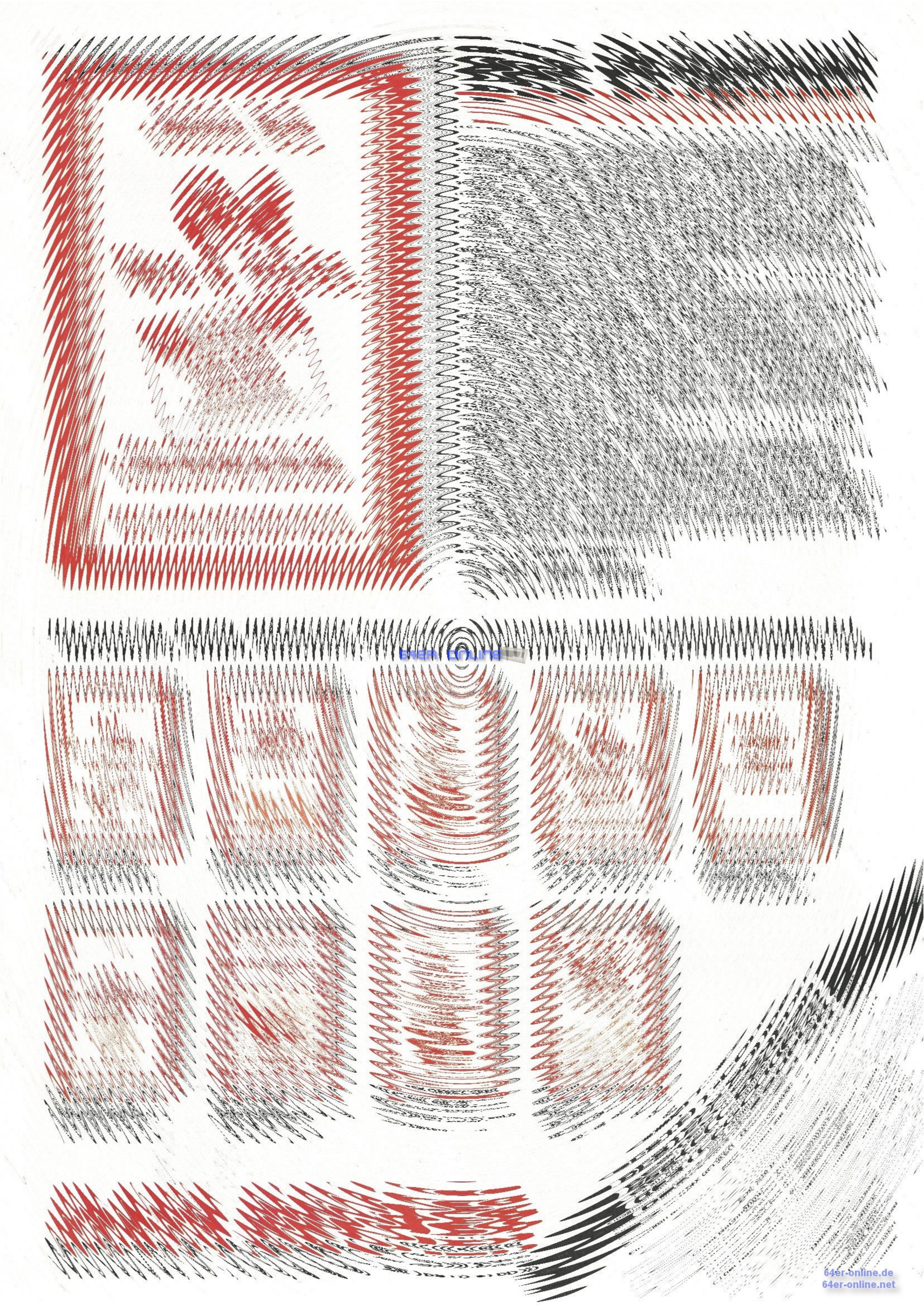
Allgemeine Daten	
Programmname:	Superbase 64
Programmart:	Datenbank
Vertrieb:	Data Becker, Merowingerstr. 30, 4000 Düsseldorf, Tel. 02 11/31 00 10
Preis:	198 Mark
Empfohlene Hardware-Ausstattung	
Computer:	C 64 oder C 128
Floppy:	1541 mit Floppy-Speeder (Turbo Access, Speeddos+, Prologic DOS)
Monitor:	Farbe Commodore 1702/1902, monochrom BMC BM12EN
Drucker:	über 1800 Mark: Fujitsu DX2100 1200 — 1800 Mark: Epson FX-85 800 — 1200 Mark: Star NL10 bis — 800 Mark: Citizen 120D
Interfaces für die Star- und Epson-Drucker:	Print-64, 315 Mark Wiesemann-Interface 92 000/G, 248 Mark Görlitz-VC-Epson-Interface 8422 mit 2-KByte-Druckerpuffer, 249 Mark PCB IF/128 von HDS Prüftechnik, 250 Mark Wiesemann-Universal-Druckerinterface 92 008/G mit 8-KByte-Druckerpuffer, 278 Mark VCI-2/2-Interface von Data Becker, 298 Mark

Tabelle 2. Das 1. Menü von Superbase 64

1. Menü	
ENTER	Eingabe eines Datensatzes in ein mit FORMAT erstelltes Layout (Rücksprung: F1 Q)
SELECT	Suche nach einem Datensatz und Anzeige auf dem Bildschirm (Rücksprung: RETURN)
FIND	Suche nach Datensätzen, die bestimmten Kriterien entsprechen und Speichern ihrer »Schlüssel« in einer Liste für eine spätere Bearbeitung (Rücksprung: F1 Q)
OUTPUT	Ausgabe von allen oder bestimmten Datensätzen (oder Teilen davon) über den Bildschirm, den Drucker oder eine Datei (Rücksprung: RETURN)
CALC	Berechnungen im aktuellen Datensatz (Rücksprung: RETURN)
REPORT	Programmgenerator zur Erstellung eines Berichtprogramms (Rücksprung: RETURN)
EXECUTE	Start des aktuellen Programms
HELP	Anzeige von Hilfsbildschirmen (Rücksprung: RETURN); Ausdruck mit CTRL/P
Sprung ins 2. Menü mit RETURN	

Tabelle 3. Das 2. Menü von Superbase 64

2. Menü	
FILE	Wechseln der aktuellen oder Anlegen einer neuen Datei (Rücksprung: RETURN)
FORMAT	Anlegen oder Ändern eines Datensatz-Layouts über bis zu vier Bildschirmseiten (Rücksprung: F1 Q)
BATCH	Berechnungen innerhalb der aktuellen Datei unter Verwendung aller oder einzelner Datensätze (Rücksprung: RETURN)
SORT	Sortieren der Datensätze nach einem oder mehreren beliebigen Feldern (Rücksprung: RETURN)
PROG	Programmerstellung zur Vereinfachung wiederkehrender Abläufe bei eigener Anwendung (Rücksprung: F1 Q)
MAINTAIN	bietet weitere Möglichkeiten über ein weiteres Menü (Rücksprung: RETURN)
MEMO	Erstellung oder Änderung von Informations- und HELP-Bildschirmen (Rücksprung: RETURN)
HELP	siehe 1. Menü
Sprung ins 1. Menü mit RETURN	



dem wird mit diesem Feld die Position des Datensatzes auf der Diskette festgehalten. Dadurch ist die außerordentlich kurze Zugriffszeit auf einen einzelnen Datensatz über den Schlüssel zu erklären. In unserem Beispiel nehmen wir den Nachnamen als Schlüsselfeld. Schreiben Sie aber bitte zunächst in die zweite Zeile des Layouts ab Spalte 14 »Adressen-Datei«. In Zeile 6 ab Spalte 7 tragen Sie dann »Nachname« ein. Dies ist der erste Feldname, der wie alle Feldnamen nicht länger als zwölf Zeichen sein darf. Anschließend drücken Sie nacheinander F1 und »k«. Mit der CRSR(rechts)-Taste geben Sie jetzt die Länge dieses Schlüsselfeldes ein (hier 18). In der Kopfzeile können Sie dabei übrigens die Länge des Feldes ablesen. Bei 18 drücken Sie RETURN.

Das Textfeld

Zwei Zeilen tiefer, wieder ab Spalte 7, geben Sie den nächsten Feldnamen »Vornamen« und ein Blank ein. Mit F1 und »t« erstellen Sie dann in der oben angegebenen Art und Weise ein Textfeld der Länge 18. Ein weiteres Textfeld mit dem Feldnamen »Strasse« und der Länge 18 folgt zwei Zeilen tiefer.

Das Zahlenfeld

Kommen wir nun zu einem Zahlenfeld. Der Feldname soll »PLZ« lauten. Nach fünfmaligem Betätigen der Space-Taste drücken Sie F1 und »n« (für number). Automatisch springt der Cursor eine Position nach rechts, um Platz für ein Vorzeichen zu machen. Mit der CRSR-Taste geben Sie nun die Anzahl der Stellen vor dem Komma (hier vier) ein. Sie könnten nun nach Eingabe eines »,« bis zu vier weitere Stellen reservieren. Beobachten Sie bei diesem Vorgang die veränderte Anzeige in der Kopfzeile. Für die Postleitzahl benötigen Sie jedoch nur vier Stellen vor dem Komma. Beenden Sie das Zahlenfeld mit RETURN.

Folgende drei Felder von jeweils 18 Zeichen Länge runden unsere Adressendatei ab: »Wohnort«, »Land« und »Notiz«. Das letzte Feld soll in einer späteren Folge dazu dienen, aus der Datei die Adressen herauszusuchen.

Die Umrandung

Gehen Sie nun mit dem Cursor an den Rand des Layouts und geben Sie F1 und »b« (für border) ein. Anschließend können Sie ein Grafiksymbol oder ein anderes Zeichen auswählen und mit Hilfe der CRSR-Tasten einen Rand zeichnen. Der Vorgang wird wieder mit RETURN abgebrochen.

Speichern des Layouts

Das Datensatzlayout kann nun nach einer Kontrolle auf Tippfehler abgespeichert werden. Dazu brauchen Sie lediglich F1 und RUN-STOP einzugeben. Bitte beantworten Sie die Frage nach doppelten Schlüsseln mit nein. Hierzu in der folgenden Ausgabe mehr.

(Gerd Wiechering/cg)

Format-Menü

Taste	Funktion
←	Widerruf der F1-Taste
F1 q	Rücksprung, ohne das neue beziehungsweise das geänderte Datensatz-Layout zu speichern
F1 RUN-STOP	Speicherung des Datensatz-Layouts; fragt vorher eventuelle Berechnungen und Konstanten ab
F1 CLR	löscht das Layout
F1 k	Schlüsselfeld mit CRSR(links/rechts)-Taste erstellen (maximal 30 Zeichen lang)
RETURN	bezeichnet das Ende eines Feldes
F1 t	Textfeld erstellen (maximal 255)
SHIFT/ RETURN	Ende eines erzwungenen Feldes. Es können nur Datensätze gespeichert werden, die eine Eintragung in diesem Feld erhalten.
F1 d	Datenfeld (Datumfeld) erstellen, sieben Zeichen für zum Beispiel 20Apr86; nach einer Erweiterung durch INST auf 11 Zeichen wird der errechnete Wochentag im Feld mit angezeigt.
F1 n	Zahlenfeld erstellen; Vorkommastelle (maximal 9) mit CRSR(rechts), »Komma« und Nachkommastellen (maximal 4) angeben.
F1 c	Konstantenfeld erstellen (maximal 30); gleiche Konstante für alle Datensätze einer Datei.
F1 C	Kalenderfeld erstellen; errechnetes Datenfeld.
F1 r	Resultatenfeld erstellen; Eintragung wird aus anderen Feldern nach einer frei definierbaren Formel (siehe F1 RUN-STOP) errechnet.
F1 +	nächsten Bildschirm anzeigen (0 bis 3)
F1 —	vorherigen Bildschirm anzeigen
F1 s	Invertieren des Bildschirms
F1 z	inverse Schrift ein- beziehungsweise ausschalten
F1 e	Löschen der Zeile, Leerzeile bleibt bestehen. Löschen des Feldes, auf dessen Markierung der Cursor steht.
F1 DEL	Löschen der Zeile; nachfolgende Zeilen rücken auf
F1 INST	Einfügen einer Zeile; nachfolgender Text rückt nach unten.
F1 b	Zeichen auswählen und Rand mit CRSR-Tasten zeichnen; mit RETURN abbrechen.
CTRL-1	Zeichenfarbe bei gedrückter CTRL-Taste durch erneutes Drücken der Zahlentaste ändern
CTRL-2	Bildschirmfarbe ändern
CTRL-3	Rahmenfarbe ändern
CTRL-p	Ausdruck des Bildschirms

Tabelle 4. Das Format-Menü von Superbase 64

Info: Data Becker, Merowinger Str. 30, 4000 Düsseldorf, Tel. (0211) 310010
Görlitz Computerbau, Lambertstr. 49, 5400 Koblenz, Tel. (0261) 2044
HDS-Prüftechnik GmbH, Maria-Eich-Str. 1, 8000 München 60, Tel. (089) 837021
Reinhard Wiesemann, Winchenbacherstr. 3-5, 5600 Wuppertal, Tel. (0202) 505077
Rolf Rocke Computer, Auestr. 1, 5090 Leverkusen 3, Tel. (02171) 2624

Sind Sie ein Geheimniskrämer?

Wenn nicht, dann haben wir ein interessantes Angebot für Sie. Gefragt sind Ihre Erfahrungen mit allen Programmen, Hardware-Erweiterungen und Druckern, die es für den C 64/C 128 gibt.

Man kann ein Computersystem auf zwei verschiedene Arten interpretieren. Zum einen ist das eine mehr oder weniger umfangreiche Ansammlung der verschiedensten Programme, Peripheriegeräte und Schnittstellen. Zum anderen ist ein Computersystem eine Einheit, bei der es darauf ankommt, daß alle Teile des Systems aufeinander eingespielt sind und zueinander passen. Die vielen Briefe, die uns täglich erreichen, zeigen, daß dieser Idealzustand leider noch längst nicht erreicht ist. Immer wieder kommt es vor, daß ein Programm, beispielsweise eine Grafikerweiterung oder eine Textverarbeitung, nicht mit dem favorisierten Drucker mangels geeignetem Interface harmonisiert. Bekannte Problemkinder sind auch die immer beliebter werdenden Floppy-Speeder, die zwar mit vielen Programmen funktionieren, aber eben nicht mit allen. Diesen Zustand wollen wir ändern. Schreiben Sie uns Ihre Erfahrungen mit den verschiedensten Teilen Ihres Computersystems (bitte genau spezifizieren). Besonders interessant für uns ist natürlich, wie Sie trotz Schwierigkeiten zu guten Ergebnissen gekommen sind. Manchmal ist es nur eine Kleinigkeit, die es ermöglicht, daß die gewünschten Resultate erlangt werden. So lassen sich beispielsweise beim Data Becker-Centronics-Interface (das von Wiesemann gefertigt wird und damit die gleichen Befehle hat) viele Anpassungsprobleme lösen, wenn man das Interface zunächst in den Linearkanal schaltet und danach fixiert. So lassen sich viele Grafik- und Textprogramme problemlos, mit allen ihren Funktionen, verwenden. Sie sehen, ein Tip muß nicht unbedingt umfangreich sein — Wirksamkeit ist gefragt.

Obwohl wir in unserer Redaktion ein wahrlich reichhaltiges Sortiment der verschiedensten Programme und Peripheriegeräte zur Verfügung haben, ist es natürlich auch uns nicht möglich, jedes Computersystem aufzubauen und alle Möglichkeiten auszuprobieren. Diese Möglichkeit haben nur Sie — unsere Leser. Warum mit diesen Informationen »hinter dem Berg halten«? Schreiben Sie uns Ihre Erfahrungen und Problemlösungen — es lohnt sich! Unter allen Zuschriften werden wir einen Überraschungspreis vergeben, für den es sich lohnt, die eigene Trickkiste etwas zu öffnen. Schicken Sie Ihren Tip unter Angabe der genauen Produktbezeichnung einschließlich der Versionsnummer (falls vorhanden) unter dem Stichwort »Computer Systeme« an:

Markt & Technik Verlag
Aktiengesellschaft,
Redaktion 64'er,
Arnd Wängler,
Hans-Pinsel-Str. 2,
8013 Haar bei München

Jetzt ist sie da: die Original 64'er -Sammelbox

Für alle Leser, die »64'er« regelmäßig kaufen, sammeln oder im Abonnement beziehen, gibt es jetzt ein interessantes Service-Angebot: Die 64'er-Sammelbox!

Mit dieser Sammelbox bringen Sie nicht nur Ordnung in Ihre wertvollen Hefte, sondern schaffen sich gleichzeitig ein interessantes und attraktives Nachschlagewerk.

Übrigens: Die Sammelbox ist nicht nur ein praktisches Aufbewahrungsmittel: Sie eignet sich auch hervorragend als Geschenk für Freunde und Bekannte zu vielen Anlässen.

Ein kompletter Jahrgang (12 Hefte) paßt in die praktische Sammel-Box! Am besten gleich bestellen!

Und so kommen Sie einfach und schnell zu Ihrer Sammelbox:

Vorbereitete Zahlkarte auf dieser Seite ausfüllen, Anzahl der gewünschten Sammelboxen, Sonderhefte oder älteren Ausgaben (s. Rückseite) angeben, Zahlkarte heraustrennen und Rechnungsbetrag beim nächsten Postamt einzahlen. Lieferung erfolgt sofort nach Zahlungseingang.

Wichtig: Es werden ausschließlich Bestellungen gegen Vorauszahlung mit Zahlkarte ausgeliefert. Ihre Bestellung wird sofort nach Zahlungseingang zur Auslieferung gebracht!

Für Mitteilungen an den Empfänger

Bestellung Leser-Service — **Wichtig:** Lieferanschrift (Rückseite) nicht vergessen!

Bestell-Nr.	Anzahl	x Einzelpreis	= Gesamtpreis
Sammelbox »64'er«		DM 14,-	DM
Sonderheft		DM 14,-	DM
Ausg. ..../1984		DM 6,50	DM
Ausg. ..../1985		DM 6,50	DM
Ausg. ..../1986		DM 6,50	DM
Zzgl. einm. Versandkostenpauschale (DM 3,-)			DM 3,-
Summe bitte auf Vorderseite übertragen.		Gesamtsumme:	DM

Hinweis für Postgirokontoinhaber:

Dieses Formblatt können Sie auch als Postüberweisung benutzen, wenn Sie die stark umrandeten Felder zusätzlich ausfüllen. Die Wiederholung des Betrages in Buchstaben ist dann nicht erforderlich. Ihren Absender (mit Postleitzahl) brauchen Sie nur auf dem linken Abschnitt anzugeben.

1. Abkürzung für den Namen Ihres Postgiroamts (PGiroA) siehe unten
2. Im Feld »Postgiroteilnehmer« genügt Ihre Namensangabe
3. Die Unterschrift muß mit der beim Postgiroamt hinterlegten Unterschriftsprobe übereinstimmen
4. Bei Einsendung an das Postgiroamt bitte den Lastschriftzettel nach hinten umschlagen

Abkürzungen für die Ortsnamen der PGiroA:

Bln W = Berlin West, Kln = Köln, Dtmd = Dortmund, Lshfn = Ludwigshafen am Rhein, Esn = Essen, Mchn = München, Ffm = Frankfurt am Main, Nbg = Nürnberg, Hmb = Hamburg, Sbr = Saarbrücken, Han = Hannover, Stgt = Stuttgart, Krh = Karlsruhe

Feld für postdienstliche Zwecke

Bedienen Sie sich der Vorteile eines eigenen Postgirokontos

Auskunft hierüber erteilt jedes Postamt

Einlieferungsschein/Lastschriftzettel
(nicht zu Mitteilungen an den Empfänger benutzen)

Gebühr für die Zahlkarte
(wird bei der Einlieferung bar erhoben)

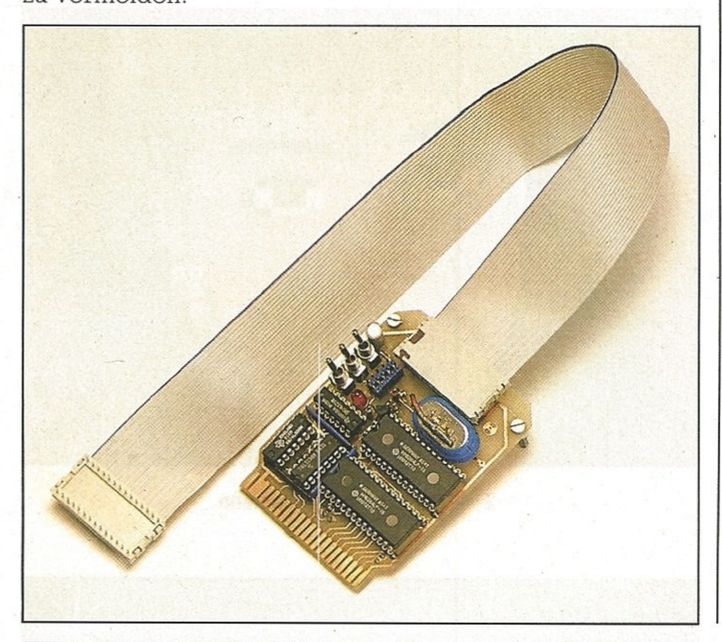
bis 10 DM — 90 Pf
über 10 DM (unbeschränkt) — 1,50 DM

Bei Verwendung als Postüberweisung gebührenfrei

Fortsetzung von Seite 43

CMOS-RAM-Platine

Hier sehen Sie die CMOS-RAM-Platine mit dem Anschlußkabel für die ROM-Simulation. Der DIL-Stecker am freien Kabelende wird anstelle eines ROMs oder EPROMs in den entsprechenden IC-Sockel gesteckt. Auf dem Bild erkennen Sie, daß der Pin 1 und die entsprechende Steckerseite markiert sind. Dies sollten Sie bei Ihren Steckern ebenfalls tun, um ein verkehrtes Einstecken zu vermeiden.



2000 Mark für Marcus Plewa

Markus Plewa ist 25 Jahre alt und studiert im fünften Semester Maschinenbau an der Technischen Hochschule in Aachen. Vor zirka acht Jahren hat er angefangen, elektronische Bauteile im Modellbau zu verwenden und daran immer mehr Gefallen gefunden. So ist die Elektronik zu einem interessanten Hobby für Ihn geworden.

Während des Zivildienstes ist Marcus Plewa mit der Computerwelt in Berührung gekommen. Auch ihn hat diese technische Errungenschaft so begeistert, daß er sich nach dem Zivildienst einen C 64 kaufte.

Im anschließenden Studium lernte er weiter C 64-Besitzer kennen. Einige beschäftigten sich damit, das Betriebssystem des C 64 zu ändern und modifizierte Zeichensätze für Drucker zu erstellen. Das ständige EPROM-Brennen erwies sich dabei bald als ein Problem.

So ist Marcus Plewa auf die Idee gekommen, ein CMOS-RAM zu bauen. Mit Hilfe seiner Elektronik-Erfahrung und den Computer-Kenntnissen ist dann die CMOS-RAM-Platine entstanden. Eine Idee und deren Umsetzung, die uns 2000 Mark wert ist.

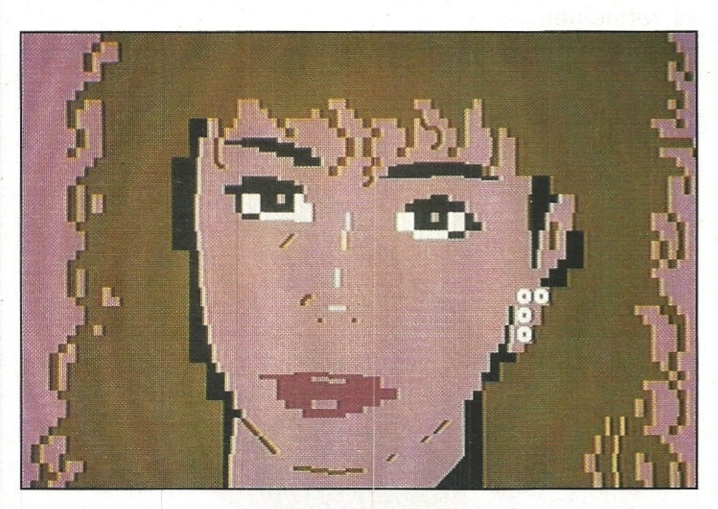
Die beste

Endrunde: Wählen Sie den Gewinne

Endlich ist es soweit! Unter all den vielen Einsendungen, die uns teilweise an den Bildschirm fesselten, ist es uns nun doch noch gelungen, eine Vorauswahl zu treffen. Manchmal konnten wir es fast nicht glauben, daß wirklich alle Einsendungen nur auf den Blockgrafik-Zeichensatz aufbauten. Die endgültige Entscheidung jedoch sollen Sie treffen. Als Belohnung für Ihre Mühe verlosen wir unter den Einsendern ein Jahresabonnement.



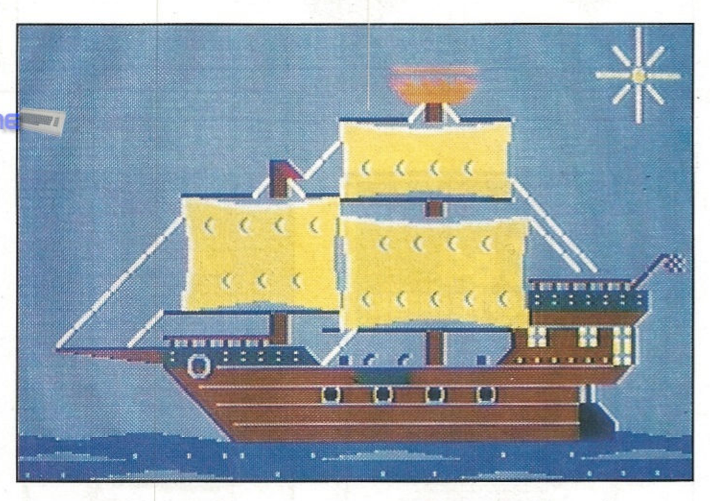
1 — Katzen von A. Kupetz



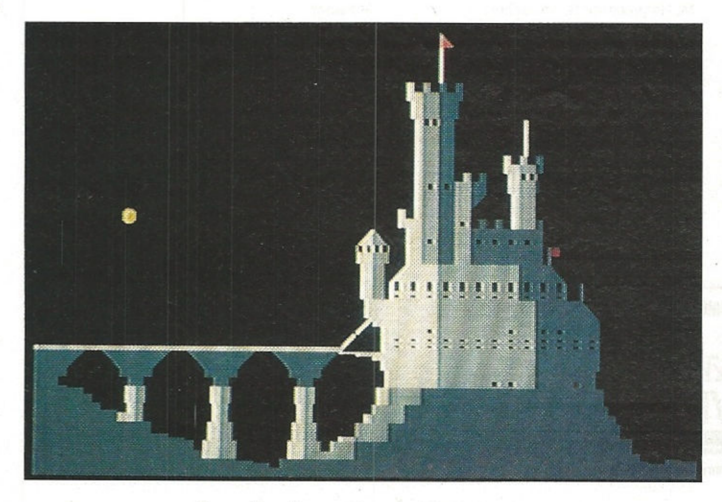
2 — Sandra von D. Reimann



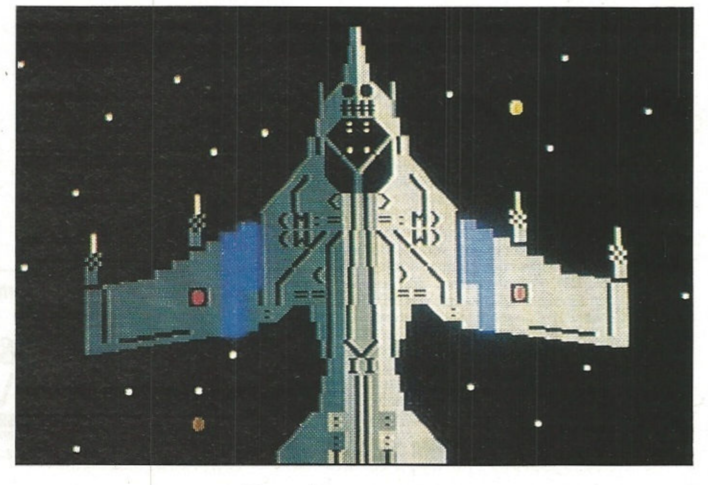
5 — Raucher von D. Tschöpe



6 — Schiff von J. Börck



9 — Frankenstein von H. Schmitt



10 — Jet von M. Trenz

n 12!

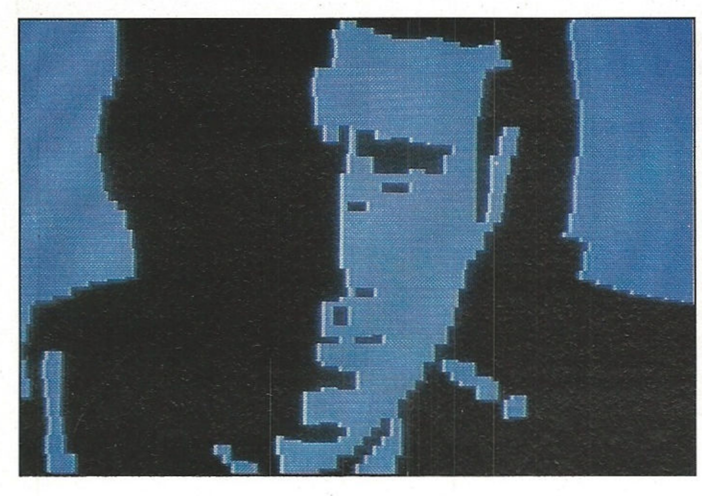
des Blockgrafik-Wettbewerbes.

Teilnahme-bedingungen:

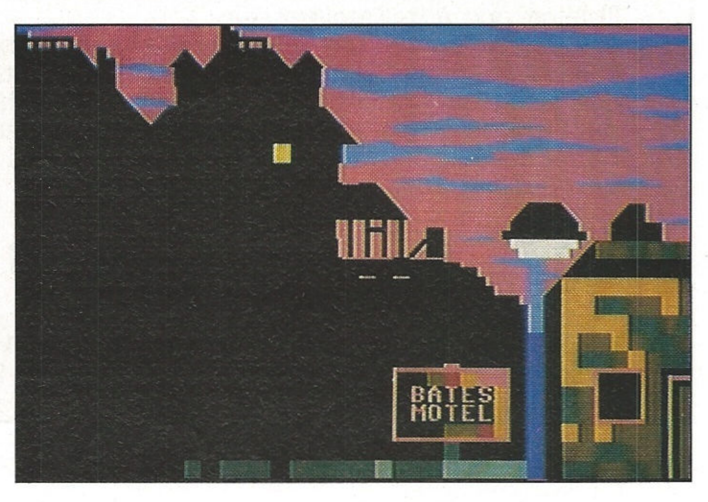
Unter den zwölf auf dieser Seite abgebildeten Blockgrafiken wählen Sie bitte diejenige aus, die Ihnen vom optischen Eindruck her am besten gefällt. Schreiben Sie die Nummer des Bildes auf eine Postkarte (bitte keine Briefe) und schicken Sie diese bis zum 31. März (Datum des Poststempels) an folgende Adresse:

Markt & Technik Verlag AG
Hans-Pinsel-Str. 2
Redaktion 64'er
Stichwort: Blockgrafikauflösung
8013 Haar bei München

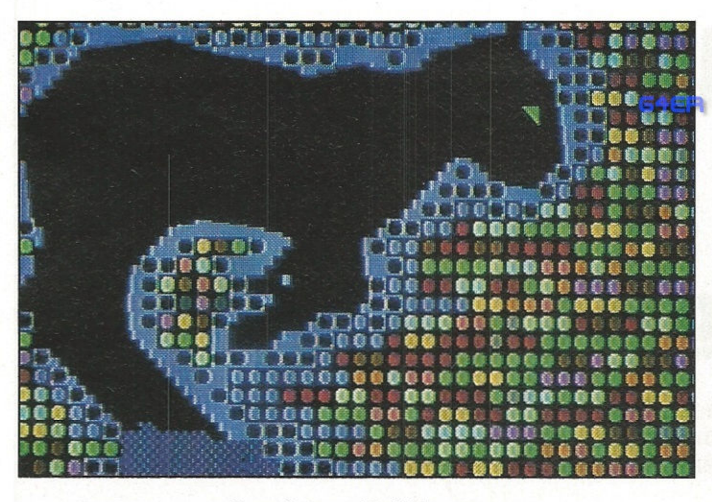
Einsendeschluß ist der 31. März 1986.



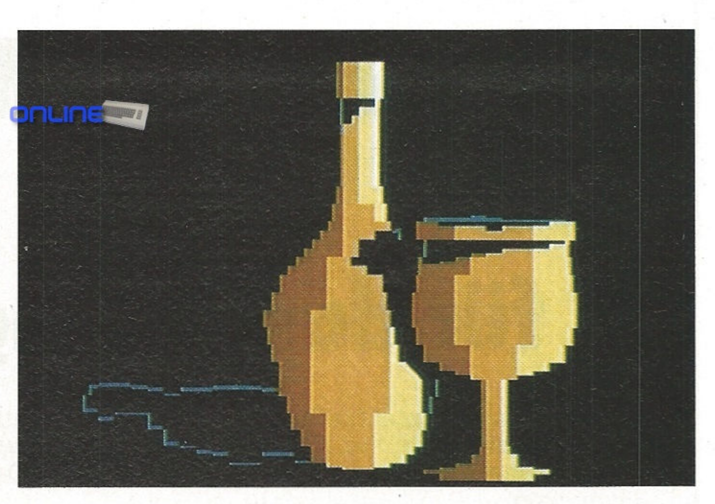
3 — Mister X von W. Randelshofer



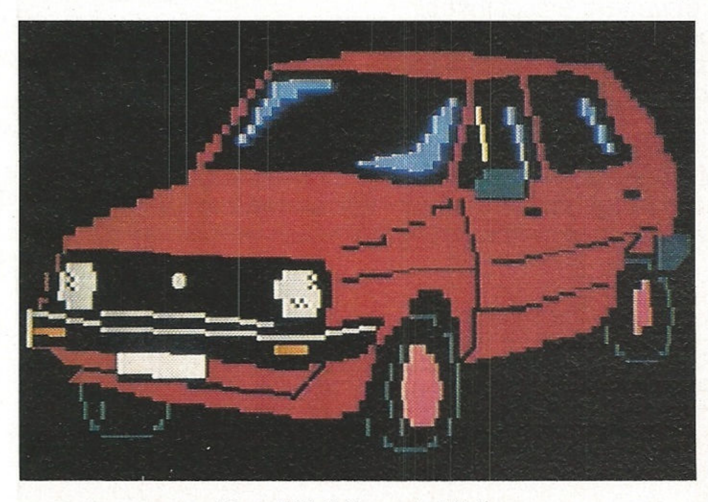
4 — Psycho von U. Hildmann



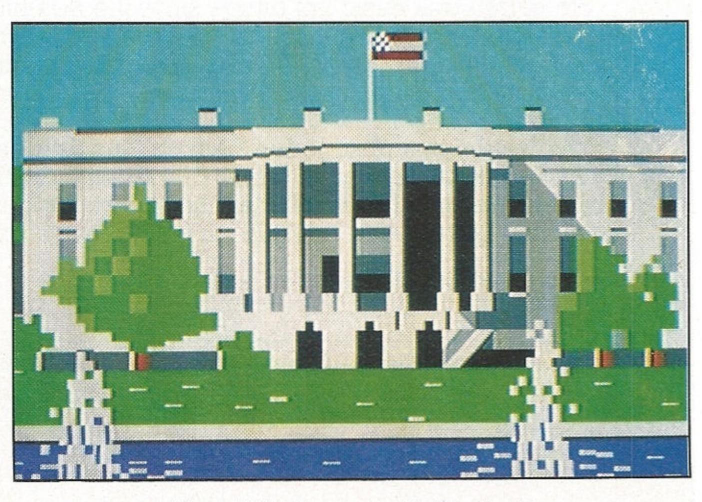
7 — Cat von U. Hildmann



8 — Stilleben von E. Hellmann



11 — VW Golf von A. König



12 — White House von C. Engelhardt

Wir suchen die Anwendung des Monats

Anwendung des Monats, was ist das? Nun, Sie haben einen Commodore 64 oder einen C 128 und versuchen diesen irgendwie sinnvoll einzusetzen. Unter einer sinnvollen Anwendung versteht die 64'er-Redaktion alles, was beispielsweise Programme im häuslichen Bereich bewirken. Es kann sich dabei um die Berechnung der Benzinkosten für Ihren Wagen handeln, um ein eigenes Textverarbeitungsprogramm gehen, sich um die Verwaltung Ihrer Tiefkühltruhe drehen oder ein ausgeklügeltes Telefon- und Adreßregister sein.

Setzen Sie Ihren C 128/C 64 mehr oder weniger beruflich ein? Auch, oder vor allem, das ist eine sinnvolle Anwendung. Sie führen die Lohn- und Gehaltsabrechnung, Ihre Lagerverwaltung, die Bestellungen auf einem Commodore-Heimcomputer durch? So spezielle Anwendungen wie die Berechnung der Statik von selbstgezimmerten Regalen, von Klimadiagrammen oder Vokabellernprogrammen für den Schulunterricht oder die Zinsberechnung bei Krediten sind ebenfalls Themen, die mehr als konkurrenzfähig sind.

Uns ist die Anwendung des Monats **500 Mark** wert. Schreiben Sie uns, was Sie mit Ihrem Computer machen:
Markt&Technik Verlag Aktiengesellschaft, Redaktion 64'er,
Aktion: Anwendung des Monats, Hans-Pinsel-Str. 2, 8013 Haar bei München.

Einmal im Monat gibt es 2000 Mark für das Listing des Monats

Diese nicht einmalige Gelegenheit sollten Sie nutzen. Wie? Schicken Sie uns Ihr bestes selbst erstelltes Programm. Bei der Art des Programms sind wir nicht wählerisch.

Sie haben ein sehr gutes (Schieß-, Knobel-, Denk-, Action-, Abenteuer-)Spiel geschrieben: einschicken!

Sie verfügen über ein komfortables Disketten-Kopier-(Sortier) Programm mit einigen außergewöhnlichen Leistungsmerkmalen: einschicken!

Sie haben das Basic um einige sinnvolle Befehle erweitert: einschicken!

Sie arbeiten mit einem selbsterstellten Textverarbeitungsprogramm, einer eigenen Tabellenkalkulation, einem semiprofessionellen Datenverwaltungsprogramm: einschicken!

Sie zeichnen und konstruieren mit einem selbsterstellten Programm in hochauflösender Grafik: einschicken!

Wir freuen uns über jeden Beitrag. Aus den besten Listings, die veröffentlicht werden, sucht die 64'er-Redaktion einmal im Monat das »Listing des Monats« aus. Alle Listings, die im 64'er abgedruckt sind, werden mit 100 bis 300 Mark honoriert. Die genaue Vorgehensweise beim Einsenden von Listings ist in dem Beitrag »Wie schicke ich meine Programme ein?« in verschiedenen Ausgaben beschrieben.

Schicken Sie Ihr Listing an: Markt&Technik Verlag Aktiengesellschaft, Redaktion 64'er,
Superchance: Listing des Monats, Hans-Pinsel-Straße 2, 8013 Haar bei München

PROGRAMM-SERVICE



Bestellungen in der Schweiz: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042/415656.
Bestellungen in Österreich: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Str. 24, 1091 Wien, Tel. 0222/481538-0
Bestellungen aus anderen Ländern bitte per Auslandspostanweisung!

Wer keine Zeit oder keine Lust hat, alle Programme selbst in mühevoller Kleinarbeit abzuschreiben, kann wieder auf den bewährten Programmservice zurückgreifen. Alle Programme, die mit dem Diskettensymbol ▪ im Inhaltsverzeichnis gekennzeichnet sind, gibt es auf Diskette.
Der Diskette liegen keinerlei Informationen bei. Lesen Sie aufmerksam die Anleitung (ob SYS-Befehle nötig sind, in welcher Reihenfolge geladen werden muß, eventuelle Sprach- oder Speicher-Erweiterungen und ähnliches mehr) in dem jeweiligen Artikel nach. Aus Aktualitätsgründen wird jeweils die abgedruckte Version angeboten.
Eventuelle systematische Fehler, die sich noch im Programm befinden können, müssen von Ihnen selbst, nach Studium des Druckfehlerteufelchens, korrigiert werden.
Der detaillierte Disketteninhalt wird mit den Seitenzahlen in der nächsten Ausgabe abgedruckt.
Wenn Sie Fragen zu den im Programm-Service angebotenen Programmen haben, wenden Sie sich an das Redaktionsteam vom 64'er-Magazin. Tel. 089/4613-202.

Das Angebot dieser Ausgabe:

Neben vielen kleinen Programmen aus unserer Tips&Tricks-Ecke haben wir diesmal etwas ganz Besonderes für Sie:
Das Hypra-Basic. Mit Hypra-Basic können Sie eine Basic-Erweiterung für Ihren speziellen Bedarf programmieren.
Besitzer eines MPS 801 oder MPS 803 können sich auf den Hardmaker freuen. Ein leistungsfähigeres Hardcopy-Programm ist kaum vorstellbar.

Diskette für den C64
Best.-Nr. L6 86 04D

DM 29,90*
sFr. 24,90/öS 299,-*

Bitte verwenden Sie für Ihre Bestellung und Überweisung die abgedruckte Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung. Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Einlieferungsschein/Lastschriftzettel
(nicht zu Mitteilungen an den Empfänger benutzen)

Gebühr für die Zahlkarte
(wird bei der Einlieferung bar erhoben)
bis 10 DM — — — 90 Pf
über 10 DM (unbeschränkt) 1,50 DM
Bei Verwendung als Postüberweisung gebührenfrei

Bedienen Sie sich der Vorteile eines eigenen Postgirokontos
Auskunft hierüber erteilt jedes Postamt

Feld für postdienstliche Zwecke

Hinweis für Postgirokontoinhaber:
Dieses Formblatt können Sie auch als Postüberweisung benutzen, wenn Sie die stark umrandeten Felder zusätzlich ausfüllen. Die Wiederholung des Betrages in Buchstaben ist dann nicht erforderlich. Ihren Absender (mit Postleitzahl) brauchen Sie nur auf dem linken Abschnitt anzugeben.
1. Abkürzung für den Namen Ihres Postgiroamts (PGiroA) siehe unten
2. Im Feld »Postgiroteilnehmer« genügt Ihre Namensangabe
3. Die Unterschrift muß mit der beim Postgiroamt hinterlegten Unterschriftsprobe übereinstimmen
4. Bei Einsendung an das Postgiroamt bitte den Lastschriftzettel nach hinten umschlagen

Abkürzungen für die Ortsnamen der PGiroÄ:
Bln W = Berlin West, Kln = Köln
Dtmd = Dortmund, Lshfn = Ludwigshafen am Rhein
Esn = Essen, Mchn = München
Ffm = Frankfurt am Main, Nbg = Nürnberg
Hmb = Hamburg, Sbr = Saarbrücken
Han = Hannover, Stgt = Stuttgart
Klrh = Karlsruhe

Für Mitteilungen an den Empfänger

Bestellung Programm-/Hardware -Service **Wichtig:** Lieferanschrift (Rückseite) nicht vergessen!

Bestell-Nr.	Anzahl	x Einzelpreis	= Gesamtpreis
Summe bitte auf Vorderseite übertragen		Gesamtsumme:	





PROGRAMM-SERVICE

Programme aus früheren Ausgaben:

Ausgabe 3/86
Bestell-Nr. L6 86 03D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)

Programm	Seite
Eingabehilfe Checksummer V3 und MSE	S. 55
Kudiplo – Funktion diskutieren und plotten	S. 58
64'er DOS – alle Funktionen der 1541 beschleunigen	S. 65
Shapes auf dem C 64 mit Demo-Programm	S. 71
Auto-Old: letzte Rettung nach »new«	S. 79
Englisch für Fortgeschrittene	S. 80
HiRes-Scrolling mit Demo-Programm und Quelltext	S. 81
1520-Plotter als Drucker	S. 82
Laufschriftgenerator – ruckfreie Laufschrift für eigene Programme	S. 83
Centronics-Interface mit Quelltext für den C 128	S. 84
View Picture – Endlich auch farbige Hi-Eddi-Bilder für eigene Programme	S. 91

Ausgabe 2/86
Bestell-Nr. L6 86 02D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)

Programm	Seite
text-transposer	S. 51
Garbage Collection: Müllabfuhr für Strings in max. 1 Sekunde	S. 53
Eingabehilfe: MSE + Checksummer	S. 57
Profiauflösung für MPS 801/803	S. 59
Software zum 64'er Eprom-Programmiergerät	S. 65
Spitzmon: Der Monitor zum Ascompiler Basic und Compilerversion	S. 69
Tips und Tricks für Profis	S. 77
Sound-Editor	S. 80
CIA: Echtzeituhr/DFÜ	S. 98
Schreiberling: Märchenstunde für Drucker MPS 801/802/803	S. 102

Ausgabe 1/86
Bestell-Nr. L6 86 01D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)

Programm	Seite
Checksummer V3	S. 54
MSE V1.0	S. 54
Datawork 1.1	S. 56
Ascompiler	S. 60
Hardcopy	S. 67
Life	S. 69
Vergleich von Programmen	S. 77
MSE-Hex-Tastatur	S. 78
Die unmögliche Uhr	S. 78
Screenlarger + Demo	S. 81
C 128 – Grafikprogramme:	
– Fensterrose	S. 131
– Spiralen	S. 133
– Box-Befehl	S. 134
IEEE-Generator	S. 147

Ausgabe 12/85
Bestell-Nr. L6 85 12D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)
Bestell-Nr. L6 85 12K Kassette
DM 29,90* (sFr. 24,90/öS 299,–)

Programm	Seite
Checksummer V3	S. 54
MSE V1.0	S. 54
Old für C 128	S. 43
Chemie-Assistent	S. 57
SMU	S. 68
Hyperscreen	S. 76
Grafik-80	S. 80
Seeschlacht	S. 93
Eprom-Automat	S. 93
Tipp-Utility	S. 99
Floppymonitor	S. 105
Auto.OBJ	S. 108
Bildsch.Langsam	S. 107
Taschenrechner	S. 107
Code-ASCII	S. 107
88-Zeichen	S. 107
Frogger	S. 106
Scroll n. unten	S. 106
Zahlenraten	S. 108
Auto-Befehl	S. 107
SWAP	S. 153
BSP-Quelltext	S. 169

Ausgabe 11/85
Bestell-Nr. L6 85 11A
DM 29,90* (sFr. 24,90/öS 299,–)

Programm	Seite
Checksummer V3	S. 54
MSE	S. 54
Koala-Painter Hardcopy	S. 39
Lyrik-Maschine (AdM)	S. 55
Hypra-Platos (LdM)	S. 61
Profiprint	S. 71
Apfelmännchen	S. 80
Block Out	S. 84
Spritekill	S. 86
Screen-Dump	S. 88
Pseudo-IRQ	S. 88
INPUT-Routine	S. 90
Synthetische Melodien	S. 95
Hypra-Ass Ergänzung	S. 96
Reassembler	S. 97
Vier Betriebssysteme	S. 105
Spiralen	S. 151
HiRes-Spiralen	S. 151

Ausgabe 10/85
Bestell-Nr. L6 85 10A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 9/85
Bestell-Nr. L6 85 09A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 8/85
Bestell-Nr. L6 85 08A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 7/85
Bestell-Nr. L6 85 07A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 6/85
Bestell-Nr. L6 85 06A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 5/85
Bestell-Nr. L6 85 05A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 4/85
Bestell-Nr. L6 85 04A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 3/85
Bestell-Nr. L6 85 03A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 2/85
Bestell-Nr. L6 85 02A
DM 29,90* (sFr. 24,90/öS 299,–)

Ausgabe 1/85
Bestell-Nr. L6 85 01A
DM 29,90* (sFr. 24,90/öS 299,–)

Sonderheft 3/86 – C 16, C 116, VC 20, Plus 4
1 Diskette für VC 20 und C 16/116:
Bestell-Nr. L6 86 S3 CD
DM 29,90* (sFr. 24,90/öS 299,–)
1 Kassette für VC 20:
Bestell-Nr. L6 86 S3 KV
DM 19,90* (sFr. 17,–/öS 199,–)
1 Kassette für C 16:
Bestell-Nr. L6 86 S3 KC
DM 19,90* (sFr. 17,–/öS 199,–)

Sonderheft 2/86 – Tips & Tricks
Bestell-Nr. L6 86 S2D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)

Sonderheft 1/86 – C 128er
Bestell-Nr. L6 86 S1D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)

Sonderheft 8/85 – Assembler
Bestell-Nr. L6 85 S8D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)
Bestell-Nr. L6 85 S8K Kassette
DM 19,90* (sFr. 17,–/öS 199,–)

Sonderheft 7/85 – Professionelle Anwendungen
Bestell-Nr. L6 85 S7D 2 Disketten
DM 34,90* (sFr. 29,50/öS 349,–)
Bestell-Nr. L6 85 S7K 4 Kassetten
DM 34,90* (sFr. 29,50/öS 349,–)

Sonderheft 6/85 – Top-Themen
Bestell-Nr. L6 85 S6 2 Disketten
DM 34,90* (sFr. 29,50/öS 349,–)

Sonderheft 5/85 – Floppy, Datasette
Bestell-Nr. L6 85 S5D Diskette
DM 29,90* (sFr. 24,90/öS 299,–)
Bestell-Nr. L6 85 S5K Kassette
DM 19,90* (sFr. 17,–/öS 199,–)

Sonderheft 4/85 – Grafik
Bestell-Nr. L6 85 S4A
DM 29,90* (sFr. 24,90/öS 299,–)

Sonderheft 3/85 – Spiele
Bestell-Nr. L6 85 S3 A 2 Disketten
DM 34,90* (sFr. 29,50/öS 349,–)

Sonderheft 2/85 – Abenteuerspiele
Bestell-Nr. L6 85 S2
DM 34,90* (sFr. 29,50/öS 349,–)

Sonderheft 1/85 – Tips & Tricks (2. überarb. Auflage)
Bestell-Nr. CB 023 Floppy-Utilities
DM 29,90* (sFr. 24,90/öS 299,–)
Bestell-Nr. CB 024 Hilfsprogramme
DM 29,90* (sFr. 24,90/öS 299,–)

* Alle Preise inklusive Mehrwertsteuer

Bedeutung der Abkürzungen

Abkürzung	Bedeutung
*LdM	Listing des Monats
*AdM	Anwendung des Monats
*SB	Simons Basic
*GV	Grundversion
*GV >	alle Speicherversionen können verwendet werden (einschließlich GV)
*3K	3-KByte-Speichererweiterung wird benötigt
*8K >	Speichererweiterung größer als 8 KByte wird benötigt
*UPB	Unterprogrammbibliothek

* Alle Preise inklusive Mehrwertsteuer.

Fehlende Hefte erhalten Sie bei:
Markt & Technik
Vertrieb 64'er
Hans-Pinsel-Straße 2,
8013 Haar


DM | Pf | für Postscheckkonto Nr. **14 199-803**
Absender der Zahlkarte
Postscheckkonto Nr. des Absenders
Empfängerabschnitt
DM | Pf
für Postscheckkonto Nr.
14 199-803
Lieferanschrift und Absender der Zahlkarte
PLZ Ort
Verwendungszweck
M&T Buchverlag
Programm-/Hardware-Service
Meine Kunden-Nr.:
PSchA | Postscheckkonto Nr. des Absenders | Postscheckteilnehmer
Zahlkarte/Postüberweisung
Die **stark umrandeten Felder** sind nur auszufüllen, wenn ein **Postscheckkontoinhaber** das Formblatt als **Postüberweisung** verwendet (Erläuterung s. Rücks.)
DM | Pf | (DM-Betrag in Buchstaben wiederholen)
für **Markt&Technik** Verlag Aktiengesellschaft
in **8013 Haar**
Postscheckkonto Nr.
14 199-803
Postscheckamt
München
Ausstellungsdatum | Unterschrift

Für Vermerke des Absenders
Postscheckkonto Nr. des Absenders
Einlieferungsschein/Lastschriftzettel
DM | Pf
für Postscheckkonto Nr. **14 199-803**
Postscheckamt **München**
für **Markt&Technik** Verlag Aktiengesellschaft
Hans-Pinsel-Str. 2
in **8013 Haar**
Postvermerk

Fortsetzung von Seite 144

STOP/RESTORE ausgelöst.

Zeile 02: Load Accumulator. In eine Speicherzelle der CPU (dem ACCUmulator) wird unmittelbar (immediate) der Wert 0 geladen.

Zeile 03: Store Accumulator in memory. Der Wert im Akkumulator wird also durch die CPU in die Adresse $D011 (dezimal 53265) gebracht. In diesem Befehl soll einmal die Arbeitsweise der CPU erklärt werden. Die CPU hat ihren Adreß-Zähler auf $1003 gestellt. Die Adresse erscheint jetzt auf dem Adreßbus des C 64. Der Speicherbaustein (in diesem Fall das RAM) stellt das Datenbyte zur Verfügung. Dieses Byte ist hier $8D. Da die CPU einen Befehl gerade beendet hat (LDA), weiß sie, daß ein neuer Befehl kommen muß. Dieses Byte ($8D) ist also ein Opcode, das heißt im Instruktionsregister (Interpreter) der CPU muß er zu finden sein. Die CPU »weiß« nach dem Erkennen des Opcodes, was sie zu tun hat: Store Accumulator heißt, den Wert im Akku zu einer Adresse zu bringen. Der Befehl $8D heißt auch, daß die Adresse absolut ist, das heißt die Adresse wird aus zwei Byte gebildet (1-Byte-Adressen beziehen sich immer auf die Zeropage, also die erste Seite im System, und das wäre nicht $8D, sondern $85).

Der Adreß-Zähler wird um eins erhöht, denn die Adresse steht logischerweise hinter dem Befehl. Die Adresse erscheint auf dem Bus, das R/W-Signal ist High (Lesen) und der Adreßmanager (siehe PLA) selektiert den Baustein (Chip Select ist Low). Das Byte, welches auf dem Datenbus der CPU angeboten wird, ist das Low-Byte (niederwertige Byte) der Adresse. Wieder wird der Adreßzähler um eins erhöht, und das High-Byte der Adresse wird geholt. Mit den zwei Bytes hat die CPU jetzt die Adresse, in die der Wert 0 aus dem Akku gespeichert (Store Accumulator) werden soll. Die CPU legt die Adresse auf den Adreßbus und die Daten (den Wert 0) auf den Datenbus. Die R/W-Leitung ist Low (Schreiben) und der Baustein selektiert. Der Befehl ist abgeschlossen, und die CPU erhöht ihren Adreßzähler, um den nächsten Befehl zu holen.

Zeile 04: Load Index X. In eine Speicherzelle der CPU (dem X-Register) wird unmittelbar (immediate) der Wert 0 geladen.

Zeile 05: Increment Memory by one. Die dem Befehl folgende Adresse wird geholt, dann der Wert der Adresse (die Daten), die CPU erhöht den Wert um eins (inkrementieren) und bringt das Ergebnis wieder zurück. Dazu braucht sie statt vier (STA) sechs Taktzyklen. Da ein Byte ja nur 8 Bit »breit« ist, kommt also nach 255 ($FF) wieder 0. Das registriert natürlich die CPU und setzt im Statusregister beim Wechsel von $FF auf $00 das Zero-Flag (Bit).

Zeile 06: Load Index Y. In eine Speicherzelle der CPU (dem Y-Register) wird unmittelbar (immediate) der Wert $64 (dezimal 100) geladen.

Zeile 07: Decrement Y by one. Der Wert des Y-Registers wird um eins erniedrigt.

Zeile 08: Branch on Result not zero. Verzweige (branch), wenn Y nicht Null ist. Dieser Befehl ist in Basic ein IF-THEN oder das FOR-NEXT in Zeile 24/26 des Basic-Listings.

Zeile 09: Increment x by one. Der Wert des X-Registers wird um eins erhöht. Die Register der CPU X und Y können also durch einen Befehl (zwei Taktzyklen) rauf- oder runtergezählt werden. Diesen Modus nennt man »implied« (mit inbegriffen), und das ermöglicht sehr effiziente und kurze Abläufe.

Zeile 10: Branch on Result not zero. Hier endet die zweite »FOR NEXT«-Schleife. Solange X noch nicht 0 ist, verzweigt die CPU wieder zur Adresse $1008. Auch hier wieder die Arbeitsweise. Nach dem Opcode $D0 kommt das Byte F5. Dieses Byte ist der Zeiger zum Verzweigen von $00 bis $7F nach vorn und von $80 bis $FF zurück. Zählt man also von $F5 bis $FF, hat man das Ziel der Verzweigung, nämlich die Adresse $1008.

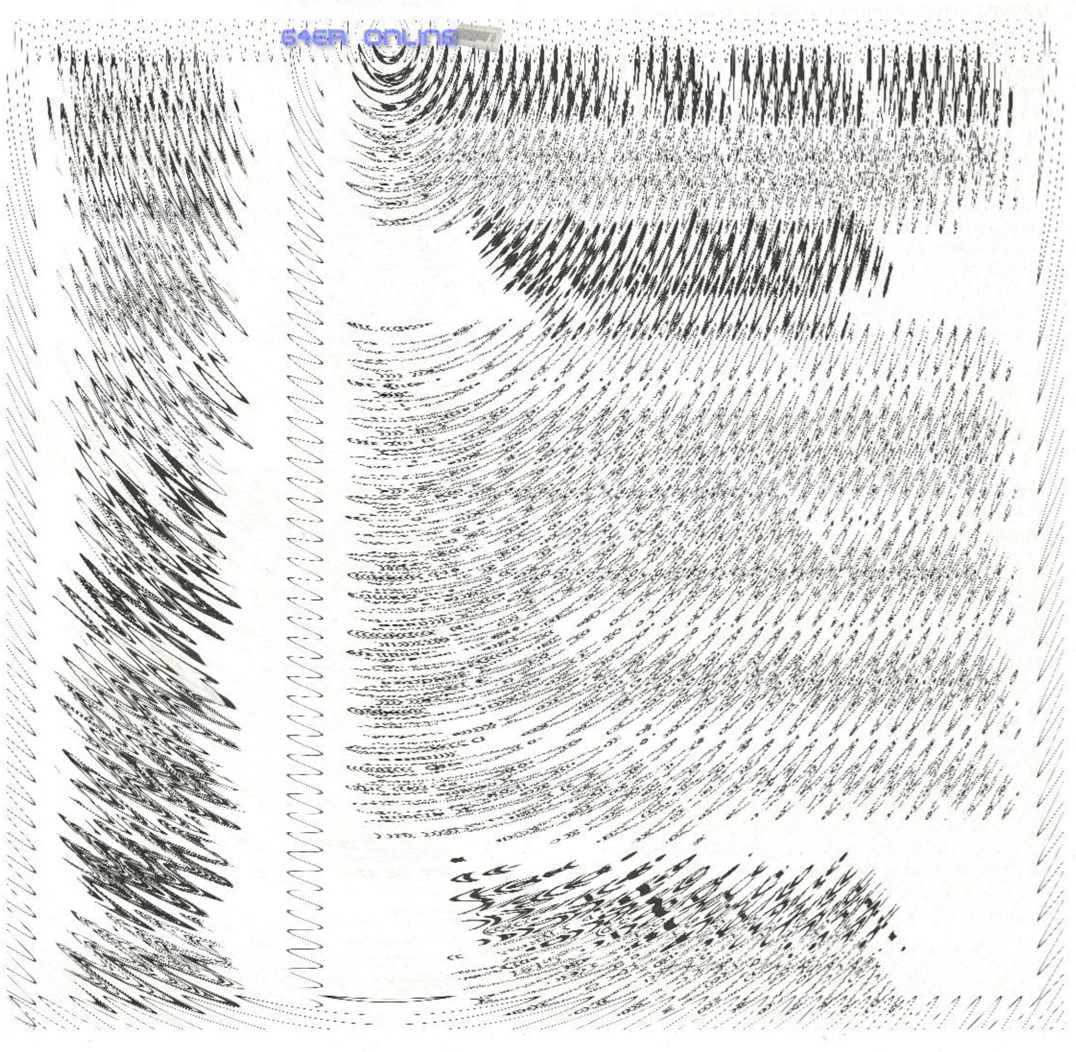
Zeile 11: Load Accumulator. In eine Speicherzelle der CPU (dem Akkumulator) wird unmittelbar (immediate) der Wert $1B (27) geladen.

Zeile 12: Store Accumulator in memory. Der Wert im Akku wird durch die CPU in das GDP-Register $D011 (dezimal 53265) gebracht. Jetzt greift der GDP wieder ins System ein.

Zeile 13: Clear Interrupt disable bit. Das Interrupt-Bit im Statusregister wird gelöscht, und die CPU muß beim Auftreten des IRQ-Impulses wieder die IRQ-Routine ausführen.

Assemblerprogrammierer winken natürlich müde ab, wenn man diese Routine, die das gleiche wie das Basic-Programm macht, zum Zeitvergleich heranzieht. Das Basic-Programm belegt im Speicher 104 Byte, das Assemblerprogramm ganze 26. Mit der Zeitersparnis von knapp vier Sekunden im Basic bei abgeschalteten GDP und IRQ braucht diese Routine zirka 37,9 Sekunden. Der Assembler durchläuft je nach Lage im Speicher bis zu 130329 Taktzyklen. Da beim C 64 in einer Sekunde die CPU 980000 Takte hat und bei unserem Beispiel weder IRQ noch GDP eingreifen, sind es 0,133 Sekunden.

Braucht die Basic-Routine also 37,9 Sekunden, hat die CPU über 37 Millionen Taktzyklen abgearbeitet. Das Assemblerprogramm war über 285mal schneller.





Bestellungen aus anderen Ländern bitte per Auslandspostanweisung!

Bestellungen aus der Schweiz bitte direkt an:
Markt&Technik Vertriebs AG
Kollerstrasse 3
CH-6300 Zug
Tel. 042/415656

Bestellungen aus Österreich bitte direkt an:
Ueberreuter Media
Handels- und Verlagsges. mbH,
Alser Straße 24,
1091 Wien
Tel. 0222/481538-0

Hardware für alle – ein neuer 64'er Leser-Service

Der Commodore 64 hat schon oft bewiesen, wie vielseitig er ist. Er läßt sich nicht nur mit Programmen, sondern auch durch so manche Hardware-Erweiterung sinnvoll nutzen und ausbauen. Dabei ist es sicherlich ein reizvoller Bestandteil des Computer-Hobbys, sich solche Erweiterungen selbst nachzubauen. Aber nicht jeder Leser verfügt über die Gelegenheit und Zeit zur Platinenherstellung. Hinzu kommt, daß es oft zu teuer ist, wegen einer bestimmten Erweiterung Investitionen von mehreren hundert Mark für eine Platinenstation zu tätigen. Die in der 64'er abgedruckten Hardware-Erweiterungen sind in drei verschiedenen Ausbaustufen zu erhalten:

1. Als Platinen
Nur Leerplatinen. Die Beschaffung der Bauteile und der Zusammenbau bleiben bei Ihnen.

2. Als Bausätze
Unsere Bausätze enthalten alle Teile, die notwendig sind, um die beschriebene Erweiterung komplett aufzubauen. Sie brauchen die Bauteile nur noch, gemäß der Anleitung im Heft, zusammenzulöten und einzubauen.

3. Als Fertiggeräte
Die Fertiggeräte sind komplett aufgebaute und geprüfte Geräte. Sie brauchen die Erweiterung lediglich noch einzubauen.

Qualität & Service

- Die 64'er Hardware hat einen hohen Qualitätsstandard. Wir verwenden nur beste Epoxid-Harz-Platinen mit Lötstopp-Lack.
- Wir verwenden nur Präzisionssockel mit gedrehten Kontakten.
- Alle Platinen werden professionell gefertigt. Wenn notwendig mit doppelseitiger Beschichtung und Löt-Durchkontaktierungen.
- Jedes Gerät, das wir versenden, wurde auf Funktionstüchtigkeit geprüft.
- Wir sind auch nach dem Verkauf für Sie da. Neben der gesetzlichen Garantie bietet unser Service- und Fertigungspartner Ihnen Hilfe und Unterstützung an.

Einbauservice

Für die Angebote 4 (Super Kernal) und 5 (64'er DOS) bieten wir einen Einbauservice an. Jeder Lieferung dieser Produkte liegt neben der detaillierten Einbauanleitung ein Angebot zum kostengünstigen Umbau Ihres C64 beziehungsweise Ihrer 1541 Floppy bei. Falls Ihr C64 keine gesockelten Bausteine besitzt, können Sie dort ebenfalls hochwertige Stecksockel einbauen lassen.

Unsere Garantie

Im Rahmen der Versand- und Lieferbedingungen unterliegen die Geräte einer Gewährleistungszeit von 6 Monaten ab Lieferung. Der Lieferung liegt eine Service-Karte bei, die Sie im Falle einer Beanstandung zusammen mit dem Gerät an die auf der Karte vermerkte Adresse schicken können. Die gleiche Karte verwenden Sie bitte bei Reparaturen nach der Garantiezeit.

Unser Angebot

Angebot 1: Expansion-Port EPROM-Platine mit 1 x 8 KByte Speicherplatz für 2732 bis 2764 EPROMS.
Beschreibung in Ausgabe 10/85
Bestellnummer: HW 010
DM 19,80* (sFr. 17,50)
Dieser Artikel wird nur als Fertiggerät angeboten.

Angebot 2: Expansion-Port EPROM-Platine mit 2 x 8 KByte Speicherplatz für 2732 bis 2764 EPROMS, mit Umschaltmöglichkeit
Beschreibung in Ausgabe 10/85
Leerplatine
Bestellnummer: HW 020
DM 24,80* (sFr. 22,–)
Bausatz mit allen Teilen:
Bestellnummer: HW 021
DM 49,80* (sFr. 43,–)
Fertiggerät, getestet, wie beschrieben:
Bestellnummer: HW 022
DM 59,80* (sFr. 51,–)

Angebot 3: EPROM Trans – Die Speichererweiterung
ROM-Speichererweiterung zum Einbau in den C 64, gleichzeitig Steckplatz für ein Original- oder ein alternatives Betriebssystem. Zwei Platinen in Epoxid-Harz-Ausführung wie in Ausgabe 10/85 beschrieben.
Leerplatine
Bestellnummer: HW 030
DM 49,80* (sFr. 43,–)
Bausatz mit allen Teilen:
Bestellnummer: HW 031
DM 119,80* (sFr. 102,–)
EPROM Trans ist nicht als Fertiggerät erhältlich.

Angebot 4: Super Kernal
Erweitertes Betriebssystem für den C 64 mit vielen neuen Funktionen, inkl. Adaptersockel, einbaufertig in den C 64.
Beschreibung in Ausgabe 11/85
Version 1: Enthält Hypra Load / DOS 5.1 / Funktionstastenbelegung / Renew / RS232
Bestellnummer: HW 040
Version 2: Enthält Hypra Load / DOS 5.1 / Funktionstastenbelegung / Renew / Super Centronics Schnittstelle
Bestellnummer: HW 041
Version 3: Enthält Hypra Load / DOS 5.1 / Funktionstastenbelegung / Renew / Hypra Save
Bestellnummer: HW 042
Version 4: Enthält Hypra Load / DOS 5.1 / Funktionstasten / Hypra Save / Centronics klein
Bestellnummer: HW 043
Jede Version kostet:
DM 39,80* (sFr. 34,–)

* Preise inkl. Mehrwertsteuer

Angebot 5: 64'er DOS
Jetzt wird das 1541 Laufwerk zum Renner. Mit wenig Aufwand beschleunigt 64'er DOS alle Funktionen des Laufwerkes. Das neue Betriebssystem für den Commodore 64 und das 1541 Laufwerk ist auf 2 Speicher-EPROMs der Sorte 2764 untergebracht und inkl. Adaptersockel einbaufertig vorbereitet.
Beschreibung in Ausgabe 3/86 (Einbauanleitung liegt bei).
Preis für beide EPROMs inkl. Adaptersockel
Bestellnummer: HW 050
DM 69,–* (sFr. 59,–)
Lieferbar ab Februar/März 1986

Angebot 6: EPROM-Programmiergerät
Programmiergerät für EPROMs der Typen: 2532, 2732, 2764, 27128, 27256, 27512. Platine aus Epoxid-Harz für Expansion-Port.
Beschreibung in den Ausgaben 12/85, 1/86 und 2/86.
Leerplatine
Bestellnummer: HW 060
DM 44,80* (sFr. 39,90)
Lieferbar ab März/April 1986
Spannungswandler
Bestellnummer: HW 061
DM 19,80* (sFr. 17,50)
Lieferbar ab März/April 1986
Betriebssoftware auf Diskette
Bestellnummer: HW 062
DM 14,80* (sFr. 13,90)
Lieferbar ab März/April 1986
Die Betriebssoftware befindet sich außerdem auf der Leserservice-Diskette der Ausgabe 2/86.
Kombinationsangebot
Leerplatine, Spannungswandler und Diskette im Paket.
Bestellnummer: HW 063
DM 69,80* (sFr. 59,50)
Lieferbar ab März/April 1986

Angebot 7: Akustikkoppler HITRANS 300 C
Mit einem Akustikkoppler öffnen Sie Ihrem Computer das Tor zur ganzen Welt. Der HITRANS 300 C stach im Akustikkoppler-Test der Ausgabe 3/86 durch die besten Übertragungseigenschaften hervor. Sie erhalten ihn bei uns als Fertiggerät, lediglich eine Blockbatterie muß eingesetzt und das Gehäuse zugeschraubt werden. Sie können den Koppler auch über ein 12-Volt-Netzteil, das in jedem Elektronikgeschäft preisgünstig erhältlich ist, betreiben. Die Bauanleitung für ein RS232-Interface finden Sie in der Ausgabe 3/85.
Preis für Akustikkoppler HITRANS 300 C (ohne Batterie)
DM 248,–* (sFr. 225,–)
Bestellnummer: HW 070
Betriebssoftware auf Diskette
Bestellnummer: HW 071
DM 14,80* (sFr. 13,90)
Die Betriebssoftware befindet sich außerdem auf der Programm-Service-Diskette des 64er-Sonderheftes SH7/85.

Bitte verwenden Sie für Ihre Bestellung immer die abgedruckte Postgiro-Zahlkarte oder einen Verrechnungsscheck. Sie erleichtern uns damit die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Inserentenverzeichnis

Impressum

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Chefredakteur: Michael Scharfenberger (sc)
Leitender Redakteur: Albert Absmeier (aa)
Redakteure: ah = Achim Hübner, aw = Arnd Wängler, bs = Boris Schneider, cg = Christine Geißler, do = Gerd Donaubauer, ev = Volker Everts, gk = Georg Klinge, hm = Harald Meyer, kn = Gottfried Knechtel, og = Markus Ohnesorg, tr = Thomas Röder,
Redaktionsassistenz: Yvonne Wilhelm (202)
Fotografie: Janos Feitser/Jens Jancke, Titelfoto: Jens Jancke
Titelgestaltung: Heinz Rauner, Grafik-Design

Layout: Leo Eder (Ltg.), Sigrid Kowalewski (Cheflayouterin), Dagmar Berninger, Willi Gründl

Auslandsrepräsentation:
Schweiz: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-41 56 56, Telex: 862329 mut ch
USA: M & T Publishing, Inc: 501 Galveston Drive, Redwood City, CA 94063; Tel. (415) 366-3600, Telex 752-351

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten werden, so muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt & Technik Verlags AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt&Technik Verlag AG verlegten Publikationen und dazu, daß Markt&Technik Verlag Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Herstellung: Klaus Buck (180)

Anzeigenverkaufsleitung: Ralph Peter Rauchfuss (126)

Anzeigenleitung: Brigitta Fiebig (282)

Anzeigenverwaltung und Disposition: Michaela Hörl (171), Liane Huber (168)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297x210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Anzeigenpreise: Es gilt die Anzeigenpreisliste Nr. 3 vom 1. Januar 1986.
Anzeigengrundpreise: 1/1 Seite sw: DM 10200,-. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala je DM 1400,-. Vierfarbzuschlag DM 3800,-. Plazierung innerhalb der redaktionellen Beiträge: Mindestgröße 1/2-Seite
Anzeigen im Computer-Markt: Die ermäßigten Preise im Computer-Markt gelten nur innerhalb des geschlossenen Anzeigenteils, der ohne redaktionelle Beiträge ist. 1/1-Seite sw: DM 7700,-. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala je DM 1400,-. Vierfarbzuschlag DM 3800,-.
Anzeigen in der Fundgrube: Private Kleinanzeigen mit maximal 5 Zeilen Text DM 5,- je Anzeige.
Gewerbliche Kleinanzeigen: DM 12,- je Zeile Text.
Auf alle Anzeigenpreise wird die gesetzliche MwSt. jeweils zugerechnet.

Marketingleiter Vertrieb: Hans Hörl (114)

Vertriebsleitung: Helmut Grünfeldt (189)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebsgesellschaft mbH, Hauptstätterstraße 96, 7000 Stuttgart 1, Telefon (07 11) 64 83-0

Erscheinungsweise: 64'er, Magazin für Computerfans, erscheint monatlich Mitte des Vormonats.

Bezugsmöglichkeiten: Leser-Service: Telefon 089/46 13-249. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement verlängert sich zu den dann jeweils gültigen Bedingungen um ein Jahr, wenn es nicht zwei Monate vor Ablauf schriftlich gekündigt wird.

Bezugspreise: Das Einzelheft kostet DM 6,50. Der Abonnementspreis beträgt im Inland DM 78,- pro Jahr für 12 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 18,- für die Zustellung im Ausland (Schweiz auf Anfrage), für die Luftpostzustellung in Ländergruppe 1 (z.B. USA) um DM 38,-, in Ländergruppe 2 (z.B. Hongkong) um DM 58,-, in Ländergruppe 3 (z.B. Australien) um DM 68,-.

Druck: E. Schwend GmbH, Schmollerstr. 31, 7170 Schwäbisch Hall

Verantwortlich: Für redaktionellen Teil: Michael Scharfenberger. Für Anzeigen: Brigitta Fiebig.

Redaktions-Direktor: Michael M. Pauly

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt&Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 089/46 13-0, Telex 522 052

Telefon-Durchwahl im Verlag:

Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089-46 13 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.

Mitglied der Informationsgemeinschaft zur Feststellung der Verbreitung von Werbeträgern e.V. (IVW), Bad Godesberg.





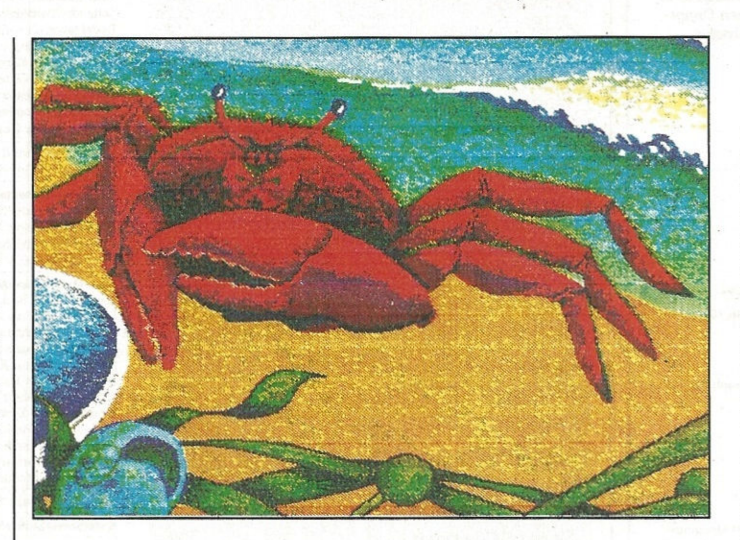
VORSCHAU



Die nächste Ausgabe erscheint am 11. 4. 86

Hannover-Messe

Vom 12.03. bis zum 19.03. findet in Hannover die CeBIT statt. Die Erwartungen für die CeBIT sind hoch, denn für den C 64 und den C 128 sind in allen Bereichen von Soft- und Hardware brandheiße Neuheiten zu erwarten. Das gilt natürlich auch für die 16-Bit-Computer, vor allem, da die offizielle Einführung des Amiga zu erwarten ist. Wir waren für Sie auf der CeBIT in Hannover und berichten aktuell vom Stand der Dinge.



Jetzt wird's bunt

Durch Farbe eröffnen sich völlig neue Dimensionen. So werden beispielsweise Texte interessanter, Grafiken schöner und Tabellen übersichtlicher. Am reizvollsten ist es natürlich, mit einem Malprogramm erstellte Bilder in Form einer Hardcopy auszugeben. Wie Sie Ihre farbigen Bilder auf das Papier bringen können und welche Drucker oder Plotter dafür empfehlenswert sind, erfahren Sie in unserem Leistungsvergleich.

Einmalige Hardcopies

Ein Problem vieler Anwender ist es, hochauflösende Farbgrafiken vom Computer auf Papier zu bringen. Von der normalen Hardcopy-Routine bis hin zu solchen für Rasterzeilen-Interrupts und Sprites bieten wir für die verschiedensten Drucker Programme an. Selbst für den Plotter VC 1520 haben wir für Sie ein superschnelles Programm in Ascomp-Basic.

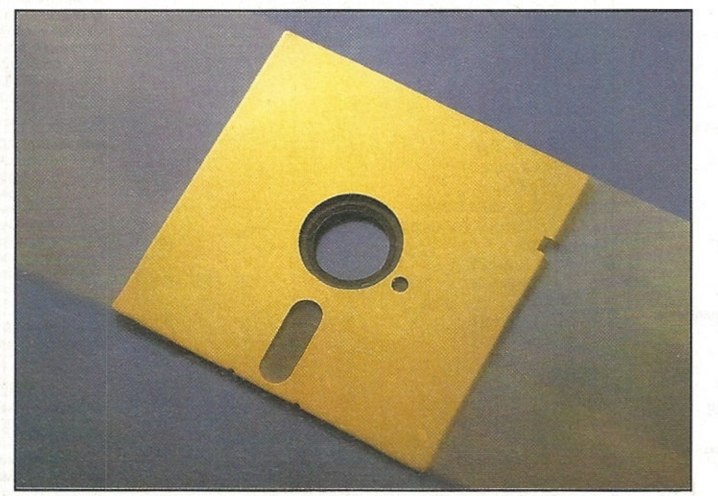
```
File Edit Search Goto Print Disk Link
VizaWrite                    Page 1 Of 1 Line 17 Pos 5
Vizawrite Classic
Der Nachfolger des für den C 64 seit lang      rogrammes
nun in einer völlig überarbeiteten Versio   745455579.25
Besondere Merkmale sind:                     C M %
- 80-Zeichen Darstellung                   7 8 9 +
- eingebautes Spell-Programm               4 5 6 -
- Deutsche Sonderzeichen                   1 2 3 /
- Window-Verarbeitung                      0 . = *
- Drop-Down Menüs wie beim Amiga
- Mail-Merge Funktion
- Rechenfunktion im Text
- Schnittstelle zur Datenverwaltung
- Automatische Seitennummerierung
- Umfangreiche Anpassungen an verschiedene Drucker
(End of Page)
```

Vizawrite 128 Classic

Was darf man von der Weiterentwicklung eines Programms, das schon auf dem C 64 zu den Besten gehörte, erwarten? Wir beantworten die Frage, wie Vizawrite auf dem C 128 aussieht. Dabei erfahren Sie, wie man mit Windows, dem Taschenrechner und der Rechtschreibhilfe arbeitet. Erstaunliches gibt es auch über die Programmbedienung zu berichten.

Hi-Eddi+ kontra Profi Painter

Gerade heutzutage ist Grafik-Software auf dem C 64 interessanter denn je. Das wird schon dadurch bestätigt, daß die Softwarefirmen immer bessere und leistungsfähigere Mal- und Zeichenprogramme auf den Markt bringen. In einem ausführlichen Test stellen wir Ihnen zwei neue Produkte vor, Hi-Eddi+ und Profi Painter. Bei beiden Programmen handelt es sich um neue Malprogramme. Hi-Eddi+ ist eine um viele Funktionen erweiterte Version von dem in Ausgabe 1/85 und Sonderheft 6/85 veröffentlichtem Programm Hi-Eddi. Profi Painter dagegen ist etwas völlig Neues. Hier wurde eine Benutzeroberfläche geschaffen, ähnlich wie bei GEM.



Disk-Wizard

Mit diesem Listing des Monats werden Sie zum Zaubermeister über Ihre Disketten! Der Disk-Wizard enthält unter anderem einen komfortablen Diskettenmonitor und eine Sortierfunktion für die Directory-Einträge. Die Einträge lassen sich auch bezüglich Name, Filetyp etc. ändern; geöffnete Files können wieder geschlossen werden. Weiterhin können Sie formatierte Disketten »wiederbeleben«, Files schützen und und und...

Grafik und Computeranimation auf Großrechnern

Großrechner werden nicht nur dazu benutzt, Programme in Hochsprachen zu erstellen. Viel interessanter ist die Anwendung zur grafischen Darstellung beliebiger Körper. Auch bewegte Grafiken farbig oder schwarz/weiß sind auf großen Anlagen recht leicht zu realisieren, denn Speicherplatz und Geschwindigkeit bereiten im Gegensatz zum C 64 kein Problem. Was aber heute auf Großrechnern möglich ist, wird vielleicht morgen schon auf Heimcomputern funktionieren. Daher wollen wir Ihnen einmal vorführen, was sich mit solchen Großrechneranlagen anstellen läßt, und wie sich dreidimensionale Grafiken erstellen lassen.

